

**Prace Naukowe
Instytutu Informatyki
Politechniki Wrocławskiej**

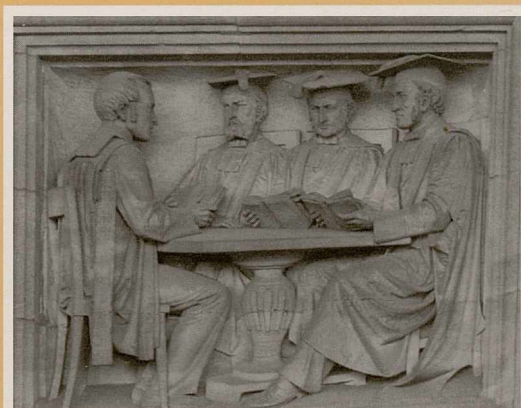
9

BAZY DANYCH

Redakcja naukowa

Zygmunt Mazur

**Oficyna Wydawnicza
Politechniki Wrocławskiej**



Prace Naukowe
Instytutu Informatyki
Politechniki Wrocławskiej

9

BAZY DANYCH

Redakcja naukowa

Zygmunt Mazur



Oficina Wydawnicza Politechniki Wrocławskiej
Wrocław 2008



Politechnika Wrocławska

Instytut Informatyki

Recenzent

Andrzej KWIECIEN

Opracowanie redakcyjne materiałów

Hanna MAZUR

Zdjęcie na okładce

Tomasz MAZUR

Wydrukowano na podstawie dostarczonych materiałów

Wszelkie prawa zastrzeżone. Żadna część niniejszej książki, zarówno w całości, jak i we fragmentach, nie może być reprodukowana w sposób elektroniczny, fotograficzny i inny bez zgody wydawcy i właścicieli praw autorskich.

© Copyright by Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław 2008

OFICyna WYDAWNICZA POLITECHNIKI WROCŁAWSKIEJ

Wybrzeże Wyspiańskiego 27, 50-370 Wrocław

<http://www.oficyna.pwr.wroc.pl>

e-mail: oficwyd@pwr.wroc.pl

ISBN 978-83-7493-443-5

Drukarnia Oficyny Wydawniczej Politechniki Wrocławskiej

SPIS TREŚCI

<i>Wstęp</i>	5
P. Pawluk, <i>Integracja i wymiana danych</i>	9
J. Glapa, A. Wilczek, <i>Korporacyjne magistrale usług</i>	17
D. Czernous, <i>Podpis elektroniczny znakiem nowej ery</i>	33
Z. Staszak, D. Jasiński, <i>Porównanie możliwości tworzenia aktywnych baz danych w SZDB Oracle i Microsoft SQL Server</i>	57
A. Gawelczyk, <i>System zarządzania bazą danych czasu rzeczywistego</i>	83
H. Mazur, Z. Górąski, <i>Gry komputerowe w programie Microsoft Access – projekt i implementacja</i>	91
R. Boborycki, A. Wilczek, <i>Charakterystyka systemów zarządzania bazami danych dla urzędów przenośnych</i>	101
R. Boborycki, A. Wilczek, <i>Porównanie wydajności wybranych komercyjnych systemów zarządzania bazami danych dla urzędów przenośnych</i>	115
H. Mazur, Z. Mazur, <i>Zarządzanie wiedzą a polityka bezpieczeństwa</i>	125
H. Mazur, Z. Mazur, <i>Ochrona zasobów komputera osobistego</i>	141
T. Mendyk-Krajewska, <i>Zagrożenia systemów sieci rozległej – skala zjawiska</i>	161
D. Żatuchin, <i>Metoda piramidy społecznej w serwisach społecznościowych</i>	175
D. Żatuchin, <i>Bazy danych w testowaniu użyteczności stron WWW</i>	183

CONTENTS

<i>Introduction</i>	5
P. Pawluk, <i>Data Integration and Exchange</i>	9
J. Glapa, A. Wilczek, <i>Enterprise Service Buses</i>	17
D. Czernous, <i>Digital Signature the Sign of New Edge</i>	33
Z. Staszak, D. Jasiński, <i>Capabilities Comparison of Creating Active Databases in Oracle and Microsoft DBMS</i>	57
A. Gawelczyk, <i>Real Time Database Management System</i>	83
H. Mazur, Z. Górawski, <i>Computer Games in Microsoft Access – Design and Implementation</i>	91
R. Boborycki, A. Wilczek, <i>Overview of Commercial Database Management Systems for Mobile Devices</i>	101
R. Boborycki, A. Wilczek, <i>Commercial Database Management Systems for Mobile Devices – Effectiveness Tests</i>	115
H. Mazur, Z. Mazur, <i>Knowledge Management and Security Policy</i>	125
H. Mazur, Z. Mazur, <i>Protection of PC Resources</i>	141
T. Mendyk-Krajewska, <i>Threats of Wide Area Network Systems – Size of Problem</i>	161
D. Żatuchin, <i>Social Pyramid Method in Social Network Services</i>	175
D. Żatuchin, <i>Databases in Usability Evaluation of Websites</i>	183

*Nauka, to nie napelnianie wiadra,
ale rozpalenie ognia*

Wiliam Butler Yeats

WSTĘP

Zachęcamy Państwa do lektury dziewiątego Zeszytu Prac Naukowych serii „*Bazy danych*”. Tym razem nie jest to zeszyt z serii *Prace Naukowe Instytutu Informatyki Stosowanej Politechniki Wrocławskiej* ale *Prace Naukowe Instytutu Informatyki Politechniki Wrocławskiej*, gdyż 1 września 2008 roku Instytut Informatyki Stosowanej został przekształcony w Instytut Informatyki. Jest to jednak nadal ta sama seria, która została zainaugurowana w 2000 roku, jeszcze w ramach Wydziałowego Zakładu Informatyki na Wydziale Informatyki i Zarządzania Politechniki Wrocławskiej. Corocznie publikowany jest jeden zeszyt z tej serii, w którym przedstawiane są wyniki prac prowadzonych głównie przez pracowników i doktorantów naszego Wydziału, związanych z szeroko rozumianą tematyką baz danych, systemów bazodanowych, bezpiecznym gromadzeniem, przesyłaniem i przetwarzaniem danych.

Kolejne prace zamieszczone w zeszycie dziewiątym dotyczą:

- integracji i wymiany danych,
- aspektów technologicznych, prawnych i psychologicznych podpisu elektronicznego,
- aktywnych baz danych,
- systemów zarządzania bazami danych czasu rzeczywistego,
- nieszablonowego wykorzystania systemu Microsoft Access do prototypowania gier typu *cRPG*,
- systemów zarządzania bazami danych dla urządzeń przenośnych,
- zarządzania wiedzą oraz polityki bezpieczeństwa informacji,
- bezpieczeństwa komputera osobistego,
- zagrożeń bezpieczeństwa systemów sieci rozległej,

- zarządzania danymi w serwisach społecznościowych,
- wykorzystania baz danych w testowaniu użyteczności stron WWW.

Konieczność wykorzystywania danych z różnych źródeł spowodowała zapotrzebowanie na systemy integracji danych, które, na przykład, umożliwią pozyskiwanie danych z różnych źródeł i przechowywanie ich w jednym miejscu w ujednoliconej postaci lub umożliwią aktualizację i wymianę informacji pomiędzy poszczególnymi źródłami danych. Autorzy dwóch pierwszych artykułów przedstawili zagadnienie integracji danych oraz związane z tym problemy i kierunki badań.

Przeprowadzanie transakcji w sposób elektroniczny i za pomocą Internetu, przechowywanie dokumentów w postaci cyfrowej wymuszają konieczność rozwoju usług szyfrowania, uwierzytelniania, niezaprzeczalności i autentyczności, zapewnienia bezpieczeństwa oraz poufności. W artykule trzecim przedstawiono niezwykle aktualne aspekty technologiczne, prawne i psychologiczne podpisu elektronicznego, wykorzystywanego obecnie przez wiele podmiotów gospodarczych na przykład w przetargach elektronicznych.

Aktywna baza danych umożliwia obserwację zdarzeń i reagowanie na takie zdarzenia. Takie możliwości ma też większość systemów baz danych ale z reguły są to mechanizmy obsługi zdarzeń elementarnych. Tymczasem systemy aktywnych baz danych umożliwiają również obsługę zdarzeń złożonych. W czwartym artykule ze zrytu dziewiątego przedstawiono przykład implementacji bazy aktywnej w dwóch systemach zarządzania bazami danych (*Oracle 11g* jak i *Microsoft SQL Server 2005*) a następnie porównano możliwości wykorzystania zdarzeń prostych oraz operatorów zdarzeniowych do konstrukcji zdarzeń złożonych w obu bazach.

W wielu dziedzinach bazach danych na przykład w systemach bankowych, telekomunikacyjnych, pomiarowych i monitorujących napływ danych jest nieustanny i operacje bazodanowe muszą się odbywać w czasie rzeczywistym. W artykule piątym omówiono systemy zarządzania bazami danych czasu rzeczywistego, problemy wynikające z ograniczeń czasowych na przetwarzanie transakcji oraz zaprezentowano wybrane rozwiązania szeregowania transakcji bazodanowych w czasie rzeczywistym.

Zastosowania baz danych i systemów zarządzania bazami danych kojarzone są na ogół z wykorzystaniem w organizacjach i przedsiębiorstwach do wspomagania procesów biznesowych. Autorzy szóstego artykułu przedstawili nieszablone wykorzystanie systemu zarządzania bazą danych *Microsoft Access* w dziedzinie gier komputerowych typu *cRPG*.

W kolejnych dwóch artykułach autorzy omówili mobilne bazy danych oraz problemy związane z ich wykorzystaniem w urządzeniach przenośnych m.in. problemy współbieżnego dostępu do danych i dostępne sposoby wymiany danych pomiędzy bazami mobilnymi. Następnie zaprezentowali trzy wybrane systemy zarządzania bazami danych dla urządzeń przenośnych: *Microsoft SQL Server 2005 Mobile*, *Oracle Lite 10g* i *Sybase SQL Anywhere 10* oraz wyniki testów wydajności przeprowadzone

na tych systemach. Na podstawie analizy uzyskanych wyników wskazano słabe oraz silne strony poszczególnych systemów.

Dane to dobro, które trzeba odpowiednio gromadzić, przetwarzać i chronić. Rozwój i zmiana technologii teleinformacyjnych sprawia, że zmieniają się również sposoby wykradania danych. Gromadzenie i przesyłanie danych w postaci elektronicznej wymusza dodatkowe działania związane z ochroną informacji oraz z zorganizowanym sposobem dokumentowania i zarządzania wiedzą. Autorzy następnych dwóch artykułów przedstawili podstawowe zagadnienia związane z zarządzaniem wiedzą i polityką bezpieczeństwa informacji oraz z ochroną zasobów komputera osobistego.

Obecnie coraz częściej atakowane są serwery i aplikacje internetowe, rośnie liczba przestępstw komputerowych, których celem jest uzyskanie korzyści finansowych. Zagrożenie stanowią też zainfekowane nakładki systemowe. Artykuł jedenasty porusza zagadnienia bezpieczeństwa danych systemów sieci rozległej. Przedstawione problemy są niezwykle ważne ze względu na skalę zjawiska.

W dwunastej pracy zeszytu dziewiątego autor porusza problem zarządzania swoimi danymi w serwisach społecznościowych. Dostępne są już serwisy (np. www.spinacz.pl czy baza członków na www.pti.wroc.pl), które umożliwiają zarządzanie swoimi danymi w sensie zarówno modyfikowania danych jak i określania, które dane którym grupom użytkowników mogą być udostępniane (na przykład znajomym, rodzinie, zalogowanym, nikomu, wszystkim itp.). Nadal jednak wiele serwisów nie posiada takiej funkcjonalności, której potrzebę coraz silniej odczuwają internauci. Obecne serwisy, w wielu przypadkach oferują użytkownikom dwie opcje: albo ujawnianie wszystkich danych albo żadnych.

Korzystając z wyszukiwarek stron w Internecie, nie zawsze jesteśmy zadowoleni z wyników wyszukiwania: czasami strony otwierają się zbyt wolno, czasami nie zawierają oczekiwanych treści lub nie odpowiada nam szata graficzna strony. Autor ostatniego artykułu przybliży tematykę testowania użyteczności stron internetowych (ang. *Web usability*), możliwość automatyzacji pewnych czynności oraz wykorzystania do tych celów odpowiedniej bazy danych.

Redaktor Zeszytu dziewiątego *Bazy danych – Prace Naukowe Instytutu Informatyki Politechniki Wrocławskiej* dziękuje Autorom za ich prace przekazane do druku w tym numerze oraz Recenzentowi za wnikliwą i rzetelną ocenę tych prac.

Zygmunt Mazur

Przemysław PAWLUK*

INTEGRACJA I WYMIANA DANYCH

Integracja danych jest coraz częściej spotykanym problemem w obecnych systemach informatycznych. Systemy wymagające „połączenia” informacji pochodzących z kilku różnych źródeł nie są już dla nikogo zaskoczeniem. W tym rozdziale zaprezentowane zostaną podstawowe informacje dotyczące zagadnienia integracji danych. Poruszone zostanie także zagadnienie ściśle związane z integracją, a mianowicie wymiana danych. Oba zagadnienia niosą bowiem ze sobą wiele wspólnych problemów.

1. WPROWADZENIE

Integracja danych to problem pozyskiwania i łączenia danych pochodzących z różnych źródeł oraz przedstawienia ich użytkownikowi w sposób jednorodny [Lenzerini 2002]. Konieczność takiego zabiegu wynika z „zalewu danych” jakiego doświadcza kadra odpowiedzialna za podejmowanie decyzji. Poczucie nieprzygotowania wynikające z braku spójnego źródła informacji dotyka ponad 80% kadry zarządzającej 0. Wielu managerów porusza się w środowisku „rozproszonym”, gdzie poszczególne fragmenty informacji rozrzucone są w wielu systemach. Uzyskanie spójnego obrazu w takim środowisku, koniecznego do świadomego podjęcia decyzji, zajmuje dużo czasu, a czasem jest niemożliwe.

Jeśli przyjrzymy się działalności takich instytucji jak banki czy chociażby operatorzy telefonii komórkowej, zauważymy z łatwością konsekwencje wspomnianego wcześniej rozproszenia informacji. Jak wiele razy spotykamy się z sytuacją, w której rachunek zostaje wysłany na błędny adres, mimo, że zmieniliśmy go w rozmowie telefonicznej z biurem obsługi klienta? Problemy tego typu są konsekwencją rozproszenia oraz słabego przepływu informacji pomiędzy systemami. Z pomocą przychodzą tutaj systemy integracji danych.

Celem tego rozdziału jest przybliżenie czytelnikom zagadnień integracji oraz wymiany danych oraz prezentacja kierunków badań prowadzonych na tym polu.

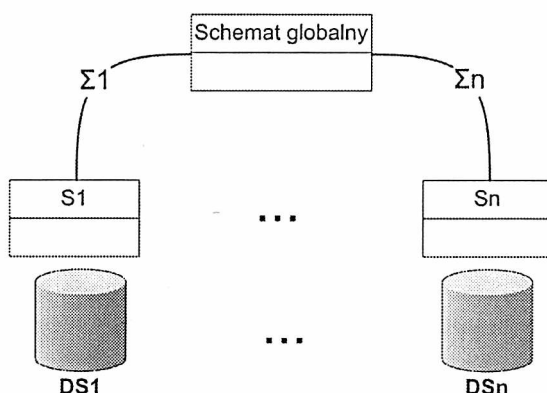
* Department of Computer Science and Engineering, York University, 4700 Keele Street, Toronto, Ontario M3J 1P3, Canada, pawluk@cse.yorku.ca

2. TEORETYCZNE PODSTAWY INTEGRACJI I WYMIANY DANYCH

Najczęściej spotykanym modelem systemu integracji danych jest model ze schematem globalnym (mediującym) oraz schematami lokalnymi. Zagadnienie mapowania czy też transformacji schematów jest problemem wspólnym dla integracji oraz wymiany danych.

2.1. INTEGRACJA DANYCH

Integracja danych, jak już było to wspomniane, jest problemem pozyskania spójnych danych z różnych, heterogenicznych źródeł oraz przedstawienia ich użytkownikowi w zadanej formie [Lenzerini 2002].



Rys. 1. Integracja danych

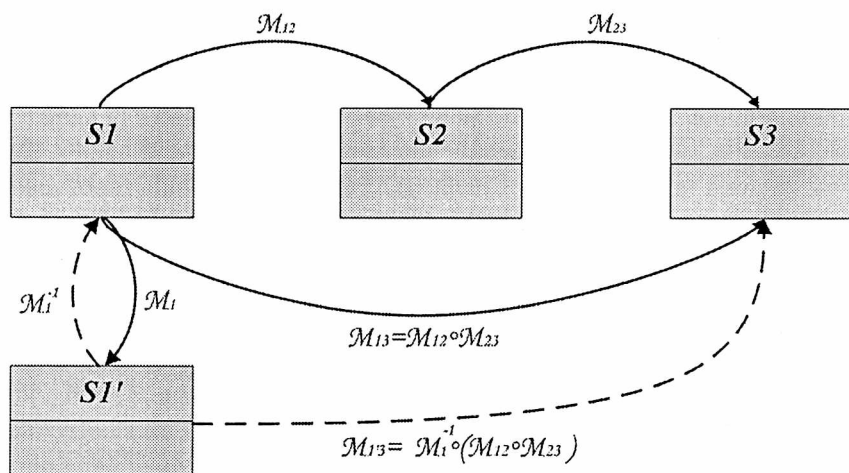
Na rysunku 1. przedstawiony został schemat integracji danych z wykorzystaniem wspomnianego wcześniej schematu mediującego. Model taki może zostać zaimplementowany na wiele sposobów. Szczególną popularnością cieszą się w tym kontekście widoki. Dwa podstawowe podejścia do implementacji tak zamodelowanych systemów z wykorzystaniem widoków to **LAV** (ang. *Local As View* – lokalny jako widok) oraz **GAV** (ang. *Global As View* – globalny jako widok).

2.2. MODEL INTEGRACJI I WYMIANY DANYCH

Prace nad transformacjami schematów prowadzone są od wielu lat, a ich główne nurty wytyczają grupy badawcze związane z IBM (przykładem mogą być prace Pho-

kiona Kolaitisa i jego grupy) oraz Microsoft (prowadzone pod kierownictwem Philipa Bernsteina).

Kolaitis w [Kolaitis 2005] opisuje w kontekście integracji i wymiany danych, w jaki sposób schematy opisane za pomocą metadanych mogą być transformowane. Bernstein wprowadza pojęcie zarządzania metadanymi [Bernstein 2003], mające stanowić narzędzie wspomagające tzw. ewolucję schematów [RaRund 1997].



Rys. 2. Ewolucja schematów. Złożenie i inwersja mapowania

Efektom prac zespołu Kolaitisa oraz Fagina było formalne opisanie metod transformacji schematów. Wprowadzili oni szereg operatorów takich jak kompozycja [FagKol 2004] czy inwersja [Fagin 2006]. Na rysunku 2. przedstawiony został przykład ewolucji schematów wykorzystujący wspomniane operatory. Punktem początkowym ewolucji przedstawionej na rysunku 2. jest transformacja schematu $S1$ (źródłowy) do schematu $S2$ (docelowego) za pomocą mapowania M_{12} . Transformacja taka, może wynikać z potrzeby wymiany informacji pomiędzy systemem opartym o schemat $S1$, a systemem korzystającym, ze schematu $S2$. W wyniku pewnych modyfikacji schemat $S2$ zostaje następnie przekształcony w schemat $S3$, co opisuje mapowanie M_{23} . Jednocześnie, aby zachować możliwość wymiany danych konieczne jest zbudowanie mapowania M_{13} . Mapowanie to jest złożeniem mapowań co zapisujemy jako $M_{12} M_{23}$. Kolejnym krokiem ewolucji jest modyfikacja schematu źródłowego $S1$ opisana mapowaniem M_1 . Zmiana taka może wynikać z różnych względów, na przykład zmiany wymagań czy poprawy błędów. Aby w dalszym ciągu możliwa była wymiana danych pomiędzy systemami konieczne jest zbudowanie mapowania M_{13} . Mapowanie to jest złożeniem mapowania M_{13} oraz mapowania odwrotnego pomiędzy schematami $S1$ i $S1'$, co zapisujemy jako $M_1^{-1} (M_{12} M_{23})$, gdzie M_1^{-1} oznacza mapowanie odwrotne do mapowania M_1 .

Prace nad formalizmami związanymi z transformacjami i dopasowywaniem schematów stoją zaledwie na początku długiej drogi. W tej sferze jest jeszcze wiele do zrobienia. Opisywane w literaturze operatory (złożenie, inwersja) nie wyczerpują wszystkich możliwości, a możliwości wykorzystania wspomnianych operatorów wydają się być szerokie. Jednym z przykładów może być formalny opis procesów ETL w hurtowniach danych.

2.3. WIDOKI

Opisany w [Halevy 2001] GAV jest oparty na prostym założeniu. Schemat globalny skonstruowany musi być w taki sposób, aby był wyrażony w terminach schematów źródłowych. Metoda ta znana jest twórcom baz danych od dawna i zapożyczona została ze sposobu definiowania widoku w standardowej, pojedynczej bazie danych. Definicja takiego widoku zawiera w sobie jednocześnie „przepis” na pozyskanie danych.

Do przetwarzania zapytań w tym podejściu stosowane są metody polegające na rozwijaniu zapytań. Metoda polecana jest dla systemów o stabilnej strukturze źródeł danych, ponieważ zmiany w schematach źródłowych niosą ze sobą nie tylko konieczność modyfikacji w systemie korzystającym z danego schematu lokalnego, ale także zmiany na poziomie globalnym.

LAV [Ullman 1997] odwraca w pewnym sensie model. Schemat globalny jest niezależny, natomiast schematy lokalne zostają opisane jako widoki w terminach schematu globalnego. Podejście takie, daje dużą swobodę w kontekście zmian na poziomie lokalnym. Dzięki niezależności schematu globalnego, zmiany w schemacie lokalnym, mają jedynie lokalne konsekwencje. Przetwarzanie zapytań w systemach opartych o LAV jest rozszerzeniem wnioskowania przy niepełnej informacji [Lenzerini2002]. Podejście to polecane jest dla systemów, o dynamicznej strukturze źródeł danych (schematów lokalnych). Wymaga natomiast stabilnej struktury globalnej, ponieważ zmiany na tym poziomie pociągają za sobą konieczność modyfikacji definicji schematów lokalnych.

Oba podejścia są często opisywane w literaturze. Ich szczegółowy opis i porównanie można odnaleźć na przykład w pracach [Lenzerini 2002] oraz [Ullman 1997].

3. MDM ORAZ ROZWIĄZANIA PRAKTYCZNE

Integracja danych w ogólności daje spójny obraz danych posiadanych przez organizację. Jednakże często spotykana jest sytuacja, w której jedynie niewielki wycinek danych jest interesujący czy istotny w podejmowaniu decyzji. Dla wielu firm będą to klient, dostawca, produkt. Dane takie nazywane są danymi kluczowymi (ang. *master*

data). Problem zarządzania danymi kluczowymi (ang. *master data management*) stał się więc tematem badań prowadzonych w wielu miejscach.

3.1. MDM W TEORII

Angielskie określenie *Master Data* czyli dane kluczowe, wskazuje na kluczowe dla organizacji biznesowe byty takie jak *klient*, *dostawca*, *adres* i im podobne. Podstawą pomysłu na zarządzanie danymi kluczowymi, czyli MDM, jest problem tak zwanych *silosów*. Dane w wielu organizacjach są rozproszone pomiędzy wieloma niezależnymi systemami, które możemy określić mianem silosów. W silosie składowane są dane, wymiana informacji pomiędzy silosami jest natomiast znacznie utrudniona. Procesy z jednego silosu nie mają dostępu do danych znajdujących się w innym silosie. Powoduje to, że utrzymanie spójnego stanu poszczególnych bytów (*klient*, *produkt*) w całym systemie staje się niemożliwe 0.

Oczywistym zyskiem z zastosowania MDM w organizacji jest utworzenie jednego, spójnego i wspólnego dla wszystkich systemów widoku danych kluczowych. Tym sposobem unikamy sytuacji, w której aktualizacja danych w jednym silosie nie będzie widoczna w innym. Znacznie łatwiejsza staje się także analiza danych.

Problemy jakie napotyka się w tym kontekście są nie różnią się znacząco od problemów spotykanych w integracji danych. Są to przede wszystkim problemy z deduplikacją (ang. *deduplication*), standaryzacją (ang. *standardization*) oraz łączeniem (ang. *merge*) danych z różnych źródeł.

3.2. DOSTĘPNE NARZĘDZIA

Przykładem rozwiązania (jednego z wielu dostępnych na rynku) jest *MDM Server* wyprodukowany przez IBM. Platforma mająca za zadanie dostarczenie zintegrowanego widoku najbardziej istotnych dla użytkownika danych [HazRun 2007]. Rozwiązanie to współpracuje z platformą integrującą dane o nazwie *IBM Information Server*, która wspomaga analizę, modelowanie i zarządzanie danymi 0. Więcej informacji na temat tego rozwiązania można znaleźć na stronie producenta [IBM1].

Konkurencyjne rozwiązanie, proponowane przez Microsoft, wspomagające integrację danych to *Microsoft SQL Server 2005 Integration Services* (dokumentacja online na stronie producenta [Microsoft 2005]). Jest to zestaw usług do tworzenia hurtowni danych. Wspomaga przeprowadzanie procesów ETL, a w ich wyniku materializację zintegrowanych danych. BizTalk [BernMelChu 2006] to kolejne narzędzie proponowane przez Microsoft wspomagające integrację danych. Jego zadaniem ma być integracja systemów EAI (ang. *Enterprise Application Integration*) oraz B2B (ang. *Business To Business*). Udostępnia on między innymi narzędzia do specyfikacji dokumentów, odwzorowań między nimi ich transformacji.

4. PODSUMOWANIE ORAZ KIERUNKI BADAŃ

W powyższym rozdziale naszkicowane zostały podstawowe zagadnienia związane z integracją oraz wymianą danych, ich teoretyczne podstawy oraz praktyczne rozwiązania dostępne obecnie na rynku. Integracja danych to zagadnienie trudne i wymagające dalszych prac, zarówno w celu zautomatyzowania procesu w jak największym stopniu jak również w innych kierunkach.

W dalszym ciągu prowadzone są prace w sferze teoretycznej mające na celu formalizację modelu i metod integracji. Dzięki tym badaniom określone zostają granice złożoności operacji związanych z integracją.

Przykładem nowych kierunków badań w zakresie integracji jest zagadnienie jakości danych zintegrowanych oraz wpływu rozmaitych czynników historycznych na wartość danych zintegrowanych dla użytkownika końcowego.

Integracja danych jest wciąż zagadnieniem nowym i stanowiącym często olbrzymi problem dla organizacji. Skutki braku zintegrowanego źródła danych dotyczą każdego z nas w czasie kontaktu z organami administracji państwowej. O ile prostsze byłoby nasze życie, gdyby za informatyzacją administracji szła jednocześnie integracja danych gromadzonych przez poszczególne jej jednostki.

LITERATURA

- [Bernstein 2003] Bernstein P.A., *Applying Model Management to Classical Meta Data Problems*. Proc. of the 2003 CIDR Conference, ACM, 2003.
- [BernMelChu 2006] Bernstein P.A., Melnik S., Churchill J.E., *Incremental Schema Matching*, Proc. of the XXXII international conference on Very Large Data Bases, September 12–15, 2006, Seoul, Korea, ACM, 2006.
- [Fagin 2006] Fagin R., *Inverting schema mappings*, ACM Transactions on Database Systems 32, 4 (2006), ACM, 2006.
- [FagKol 2004] Fagin R., Kolaitis P.G., Tan W.-C., Popa L., *Composing schema mappings: second-order dependencies to the rescue*. PODS '04: Proc. of the XXIII ACM SIGMOD-SIGACT-SIGART Symposium on Principles of database systems (New York, NY, USA, 2004), ACM, 2004.
- [Halevy 2001] Halevy A., *Answering queries using views: A survey*, Very Large Data Base Journal, 10, 4 (2001), VLDB Endowment, 2001.
- [HazRun 2007] Hazlewood S., van Run P., *Master Data Management in action: Maintaining Customer Truth Across The Enterprise*, prezentacja z CASCON 2007.
- [IBM1] IBM *Information Integration*, <http://www-306.ibm.com/software/data/integration/>.
- [IBM2] IBM White Paper, *Master data management: looking beyond the single view to find the right view*, http://www-935.ibm.com/services/us/cio/pdf/leverage_wp_mdm_beyond_single_view.pdf
- [Kolaitis 2005] Kolaitis P.G. *Schema mappings, data exchange, and metadata management*, PODS '05: Proc. of the XXIV ACM SIGMOD-SIGACT-SIGART: symposium on Principles of database systems (2005), ACM, 2005.

- [Lenzerini 2002] Lenzerini M., *Data integration: a theoretical perspective*. In: PODS '02: Proc. of the XXI ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems (2002), ACM, 2002.
- [Microsoft 2005] Microsoft, *Microsoft SQL Server Integration Services*, SQL Server 2005 Books Online, <http://msdn2.microsoft.com/en-us/library/ms141026.aspx>.
- [Pankowski 2007] Pankowski T., *Integracja danych w teorii i praktyce – przegląd problemów i rozwiązań*. Bazy Danych: Nowe Technologie (2007), WKŁ, 2007.
- [RaRund 1997] Ra Y., Rundensteiner E.A., *A Transparent Schema-Evolution System Based on Object-Oriented View Technology*, IEEE Transactions on Knowledge and Data Engineering 9, 4 (1997), IEEE, 1997.
- [Ullman 1997] Ullman J.D., *Information integration using logical views*, Proc. of the VI International Conference on Database Theory (ICDT '97), Vol. 1189 of Lecture Notes in Computer Science (1997), Springer, 1997.

DATA INTEGRATION AND EXCHANGE

The data integration problem is a common issue in many information systems. They require “merge” of data from different sources to provide one unified view. The work presents basic concepts of data integration and exchange, and presents some practical solutions.

*ESB, korporacyjna magistrala usług
integracja systemów informatycznych,
Mule, ServiceMix*

Jakub GLAPA
Artur WILCZEK*

KORPORACYJNE MAGISTRALNE USŁUG

W wielu realizowanych obecnie projektach informatycznych pojawiają się zadania dotyczące łączenia różnych systemów informatycznych. Wraz ze wzrostem liczby nieudanych projektów integracyjnych oraz rosnących kosztów utrzymania zintegrowanych systemów zaczęto szukać rozwiązań, które ułatwiły by ten proces. Jednym z takich rozwiązań jest zastosowanie korporacyjnej magistrali usług – ESB (ang. *Enterprise Service Bus*). W niniejszym artykule scharakteryzowano funkcje i zastosowania, przy integracji aplikacji, narzędzi ESB.

1. WPROWADZENIE

Informatyzacja wielu sektorów biznesowych w ostatnich latach doprowadziła do znacznego polepszenia usług świadczonych przez różnego typu przedsiębiorstwa. Wiele procesów biznesowych jest dziś wspomaganych przez specjalizowane systemy informatyczne. Procesy te są często bardzo skomplikowane i obejmują wiele niezależnych aplikacji. Dotyczy to szczególnie międzynarodowych korporacji, które swym zasięgiem obejmują niemal cały świat, zacierając przy tym wszelkie granice zarówno społeczne jak i kontynentalne. W związku z częstymi przejęciami i fuzjami przedsiębiorstw, uwidoczniły się problemy łączenia ze sobą istniejących niezależnych systemów, które dotychczas funkcjonowały w odseparowanych środowiskach. Problem integracji aplikacji istnieje od dawna, można powiedzieć, że pojawił się w momencie pierwszych prób łączenia dwóch różnych systemów, które powinny zacząć komunikować się ze sobą, oraz wymieniać dane. Jednak dopiero w ostatnich latach nabrał on szczególnego znaczenia z powodu licznych porażek, jakie ponoszono na tym tle. Według [Linth 2000] koszty integracji przeznaczane przez firmy i przedsiębiorstwa plasujące się na liście „Fortune 1000” sięgają blisko 100 miliardów dolarów rocznie. Szacunki Forrester Research mówią, że to około 30 procent

* Instytut Informatyki, Wydział Informatyki i Zarządzania Politechniki Wrocławskiej, 50-370 Wrocław, Wybrzeże Wyspiańskiego 27, artur.wilczek@pwr.wroc.pl

wszystkich środków poświęcanych na szeroko pojęte technologie informacyjne (ang. *Information Technology*). Co gorsze 70 procent wszystkich projektów integracyjnych w roku 2003 poniosło porażkę [WWW1].

Tego rodzaju fakty dość wyraźnie potwierdzają, że integracja aplikacji nie jest tematem łatwym, oraz że nie powinna być bagatelizowana przy projektowaniu nowych systemów.

1.1. POTRZEBA INTEGRACJI

Rywalizacja na dzisiejszym rynku wymusza na firmach nieustające współzawodnictwo oraz konieczność szybkiej reakcji na zmieniające się wymagania. Przedsiębiorstwa opierające swoją działalność na systemach informatycznych są zmuszone do ciągłego wprowadzania zmian w swoich aplikacyjnych, a częsta konieczność wdrażania nowych systemów i podłączania ich do już istniejących powoduje duże problemy i koszty.

Modnym słowem w ostatnich latach stało się słowo – integracja. Kluczem do odniesienia sukcesu coraz częściej nie jest już oferowanie nowych usług poprzez wdrażanie nowych systemów, ale oferowanie tych usług korzystając z już istniejących zasobów, przy jak najmniejszych nakładach finansowych.

Jak już wspomniano koszty ponoszone przy integracji dwóch systemów mogą być bardzo wysokie, a wielu przypadkach bardziej opłaca się stworzenie nowego systemu od nowa. Nie zawsze jednak można sobie na to pozwolić.

Dzisiejsze korporacje eksploatują bardzo często setki różnych aplikacji, spełniających mniejszą lub większą rolę dla sprawnego funkcjonowania przedsiębiorstwa [Hohpe 2003]. Aplikacje takie powstają w wyniku specyficznych potrzeb, czasami są tworzone tylko tymczasowo, ale często pozostają już na stałe. Z powodu rozległości podejmowanych działań biznesowych korporacje nie są w stanie centralnie kontrolować wszystkich działań każdej jednostki. Nietrudno znaleźć dziś firmy, które mają po kilkanaście różnych stron internetowych, trzy instancje systemu klasy ERP i niezliczoną ilość wewnętrznych dedykowanych aplikacji. Bardzo często są to systemy spadkowe, które ze względu na swoje specyficzne cechy muszą być utrzymywane, a jeszcze częściej są to kombinacje takich starych systemów z nowo powstałymi.

Powodów takiego stanu jest wiele. Przede wszystkim tworzenie systemów biznesowych jest skomplikowane i nigdy nie będziemy w stanie objąć jedną aplikacją wszystkich zgłaszanych wymagań. Twórcy systemów klasy ERP¹ odnieśli pewne sukcesy na tym polu, niestety nawet tacy dostawcy jak SAP, Peoplesoft czy Oracle nie są w stanie sprostać wszystkim wymaganiom.

¹ ang. *Enterprise Resource Planning* – Planowanie zasobów przedsiębiorstwa.

Konkurencja na rynku wymusiła na dostawcach rozwiązań informatycznych oferowanie różnorodnych produktów skupionych na konkretnym zastosowaniu biznesowym. Przedsiębiorstwa mają możliwość wyboru aplikacji, która najlepiej spełnia ich wymagania i nie muszą być uzależnione o jednego dostawcy.

Użytkowników końcowych nie interesuje zbytnio przebieg danego procesu biznesowego. To czy jest to zmiana danych osobowych u operatora GSM, czy zamówienie produktu w sklepie internetowym, które pociągają za sobą konieczność skorelowania ze sobą działań dwóch lub sześciu systemów nie ma dla nas większego znaczenia. Jednak od strony biznesowej jest to olbrzymie wyzwanie wymuszające integracje wielu zupełnie różnych aplikacji w celu zrealizowania danego zadania.

1.2. TRADYCYJNE PODEJŚCIE DO INTEGRACJI

Problemy pojawiające się przy integracji aplikacji nie są zagadnieniem nowym. Pewien dorobek w tej dziedzinie wniosło podejście określane jako Korporacyjna Integracja Aplikacji (EAI, ang. *Enterprise Application Integration*).

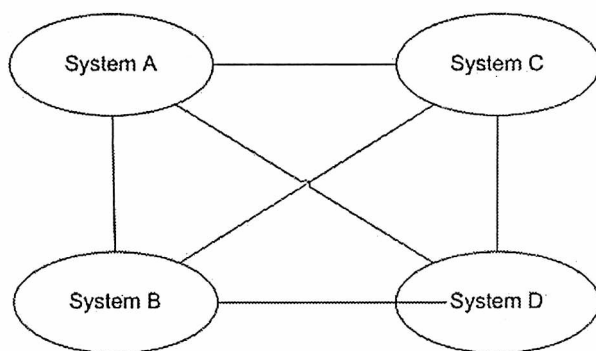
Największym wyzwaniem EAI była i jest heterogeniczność systemów i aplikacji. Większość z nich rezyduje na różnych systemach operacyjnych, wykorzystuje różne bazy danych, komunikuje się po odmiennych protokołach i najczęściej napisana jest w różnorodnych językach. Systemy spadkowe (ang. *Legacy systems*) eksploatowane przy braku wsparcia dostawców i specjalistów powodują dodatkowe problemy architektem integracyjnym.

Praca programisty przy integrowaniu systemów, polega w głównej mierze na pisaniu licznych adapterów, konektorów oraz na ustalaniu wspólnych struktur danych pomiędzy łączonymi systemami. W miarę upływu czasu i zmieniających się wymagań rozwiązań tych przybywa, aż stają się trudne do kontrolowania i administracji. Na przestrzeni lat opracowano kilka modeli integracji, najważniejsze z nich zostały przedstawione w dalszej części pracy.

Najczęściej stosowane architektury integracji aplikacji to [Binil 2008]:

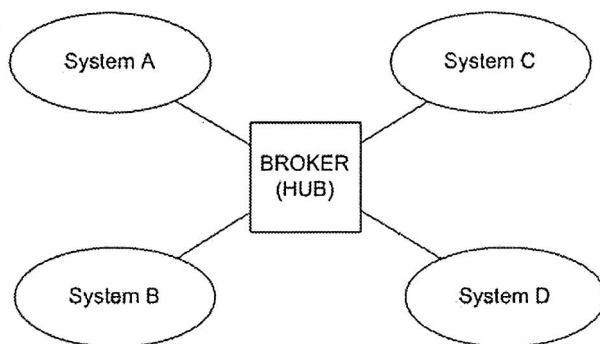
- Punkt-Punkt (ang. *Point-to-Point*),
- Model centralnego węzła (ang. *Hub-and-Spoke*),
- Magistrala wiadomości (ang. *Message Bus*).

Architektura Punkt-Punkt to powszechny sposób łączenia systemów ze sobą. Dla każdej z linii widocznych na rysunku 1, potrzeba specjalizowanej implementacji połączenia z konkretnym systemem, spowodowanej różnym środowiskiem działania lub odmiennym protokołem komunikacyjnym. Największym problem tego modelu jest rosnący koszt integracji w miarę rozrastania się systemu. W przypadku podłączania nowego systemu będziemy potrzebować kolejne n implementacji połączeń przy założeniu, że mamy n systemów. W wyniku zastosowania takiego podejścia powstaną także silne zależności (ang. *tight-coupling*) pomiędzy łączonymi systemami.



Rys. 1. Architektura Punkt-Punkt

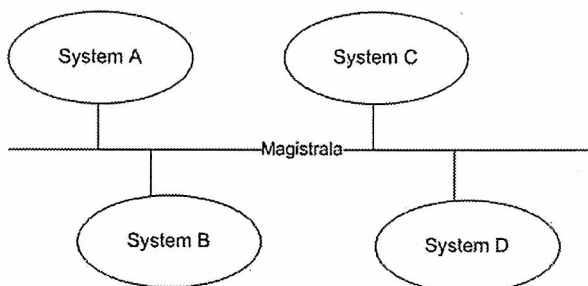
Architektura (model) centralnego węzła, zwana w nomenklaturze angielskiej również *message broker*, była i jest najpowszechniej wykorzystywanym rozwiązaniem działającym w środowisku biznesowym. Świadczy o tym chociażby gama dostępnym produktów oferowanych przez największych dostawców, tj. IBM, Oracle, Tibco. W tym podejściu każdy z systemów podłączony jest nie bezpośrednio do siebie, ale do jednego centralnego systemu (broker, hub), który koordynuje przesył danych pomiędzy systemami. Takie rozwiązanie redukuje liczbę potrzebnych połączeń pomiędzy aplikacjami w stosunku do modelu punkt-punkt (rys. 2). Liczba koniecznych połączeń została ograniczona do 4, a podłączenie kolejnych systemów ma stały koszt równy jeden. Jednak i ten model nie jest idealny, ponieważ jest podatny na uszkodzenia, w przypadku awarii systemu centralnego istnieje wysokie ryzyko awarii całego naszego systemu.



Rys. 2. Architektura centryczna

W architekturze magistrali wiadomości warstwa MOM (ang. *Message-Oriented Middleware*) jest warstwą przesyłu wiadomości, która pozwala na operowanie różnym heterogenicznym platformom dzięki dostarczeniu wspólnego dla wszystkich interfejsu

komunikacyjnego. Najważniejszymi pojęciami są wiadomość i kolejka, które służą kolejno przenoszeniu danych i magazynowaniu nieobsłużonych jeszcze żądań. Oprogramowanie MOM jest zazwyczaj zorientowane na asynchroniczne wywołania.



Rys. 3. Architektura magistrali

1.3. ARCHITEKTURA ZORIENTOWANA NA USŁUGI

Akonim SOA (ang. *Service Oriented Architecture*) oznacza koncepcję tworzenia systemów informatycznych, w której główny nacisk kładzie się na definiowaniu usług, które spełnią ściśle wymagania biznesowe. Zalecenia architektury zorientowanej na usługi (SOA) dotyczą zarówno organizacji procesów biznesowych przedsiębiorstwa jak i zagadnień technicznych, mają na celu lepsze powiązanie biznesowej strony organizacji z jej zasobami informatycznymi. Zachowanie tych zaleceń pozwala na takie zaprojektowanie architektury systemu, by był on podatny na liczne modyfikacje, tak częste w dzisiejszych czasach.

Usługi w architekturze SOA bazują na pewnych abstrakcyjnych interfejsach udostępnianym potencjalnym konsumentom [Binil 2008]. Klient takiej usługi nie musi znać jej szczegółów implementacyjnych, pociąga to za sobą możliwość współdzielenia komponentów i usług z różnych innych platform i środowisk.

2. KORPORACYJNA MAGISTRALA USŁUG

Korporacyjna magistrala usług – ESB (ang. *Enterprise Service Bus*) jest kolekcją usług pośredniczących, zapewniających możliwość integracji aplikacji. Te usługi zawarte są w rdzeniu ESB i wykorzystywane są przez integrowane aplikacje.

ESB wykorzystuje usługi sieciowe oraz standardy stosowane dotąd na przykład przy korzystaniu z brokerów EAI [Chapp 2004]. W przypadku ESB istotne jest jednak wykorzystanie otwartych standardów i odejście od architektury gwiazdистой (centrycznej) co pozwala uzyskać znaczący skok skalowalności usług oraz zwiększenie ich ponownego wykorzystania.

Krótko mówiąc ESB jest produktem technologicznym, który rozwiązuje problemy integracyjne dzięki stosowaniu przesłanek architektury zorientowanej na usługi. Wykorzystując istniejące już usługi można tworzyć zupełnie nowe, poprzez odpowiednie skonfigurowanie szyny do pełnienia zadanej funkcjonalności. W ten sposób, dzięki wykorzystywaniu istniejących już zasobów, obniżane są znacząco koszty integracji.

2.1. FUNKCJE ESB

Szyny ESB oferowane przez różnych dostawców oferują różnią się co do oferowanych funkcji. Pewien zbiór cech, jakie szyna usług powinna posiadać sformułowano w pracy [Dirk 2008] są to przede wszystkim:

- **Przezroczystość umiejscowienia** (ang. *location transparency*).
Szyna usług powinna umożliwiać rozluźnienie zależności między konsumentami (ang. *service consumers*) a dostawcami usług (ang. *service providers*). Jednym z głównych manifestów architektury SOA jest ograniczenie silnych zależności (ang. *tight coupling*) pomiędzy aplikacjami. Korzystający z usługi nie musi wiedzieć gdzie znajduje się dostawca, dzięki temu przy zmianie miejsca danej usługi nie będzie to miało wpływu na konsumenta. Oprogramowanie, w tym przypadku ESB samo zadba o przekierowanie żądania w nowe miejsce bez potrzeby angażowania strony inicjującej.
- **Konwersja protokołu komunikacyjnego** (ang. *transport protocol conversion*).
Szyna musi być w stanie dokonać odpowiedniej konwersji np. gdy dwa systemy komunikują się przy różnych protokołach, przykładowo SOAP (ang. *Simple Object Access Protocol*) na JMS (ang. *Java Messaging Service*).
- **Transformacja formatu wiadomości** (ang. *message transformation*).
Nie ma praktycznej możliwości, aby dwie różne aplikacje miały wspólny format danych, być może jest on w obydwu przypadkach XML-em, ale jego struktura może być zupełnie inna. Musi istnieć odpowiednia logika, która poradzi sobie ze stosowaną transformacją jednego formatu w drugi. Najczęściej są to narzędzia, XSLT, które dokonują odpowiednich zmian.
- **Trasowanie wiadomości** (ang. *message routing*).
Następną fundamentalną funkcjonalnością jest zdolność do wyznaczania tras komunikatów. W średniej wielkości środowisku przy kilkunastu różnych systemach, skierowanie danej wiadomości w odpowiedni miejsce jest bardzo odpowiedzialnym zadaniem. Zazwyczaj jest to realizowane przez skomplikowaną logikę, lub osobny podsystem ze zbiorem licznych reguł odpowiedzialnym za przekierowywanie żądań. Wyróżniamy kilka rodzajów trasowań, jak na przykład oparty o zawartość (ang. *content-based routing*).
- **Wzbogacanie wiadomości** (ang. *message enhancement*).
Bardzo częstym scenariuszem przy integracji systemów jest konieczność wzbogacenia jakiejś wiadomości o konkretne dane zanim dotrze do odbiorcy. Może

to być na przykład pobranie danych z bazy lub konwersja danych monetarnych, metrycznych. Różnica pomiędzy transformacją, a jej wzbogacaniem polega na tym, że transformowanie ma do czynienia z danymi już dostępnymi, a wzbogacenie ma do czynienia z informacjami które dopiero zostaną dodane.

- Bezpieczeństwo (ang. *security*).
Z powodu bardzo częstej interakcji szyny ESB z krytycznymi elementami infrastruktury biznesowej, musi ona oferować mechanizmy uwierzytelniania i autoryzacji aplikacji wykorzystujących dane środowisko. Bezpieczeństwo przesyłanych musi także być zapewnione np. przez szyfrowanie. Dzięki takiemu podejściu otrzymujemy jednolitą politykę, gwarantującą jednakowy poziom bezpieczeństwa dla wszystkich elementów współdziałających systemów.
- Monitorowania i zarządzanie (ang. *monitoring and management*).
Zapewnienie odpowiednich możliwości monitorowanie działania szyny powinno umożliwić szybkie podjęcie działań w razie wystąpienia ewentualnych problemów. Naturalne wydaje się włączenie do oprogramowania szyny ESB osobnego modułu sprawującego pieczę nad działającymi systemami.

3. SZYNY ESB OPEN SOURCE

Oprócz szeregu komercyjnych narzędzi klasy ESB dostępne są także rozwiązania Open Source. Wybrane rozwiązania tej klasy zostaną omówione w dalszej części pracy.

3.1. MULE

Mule² to lekki framework przesyłania wiadomości, który pozwala na szybkie podłączenie aplikacji w celu udostępnienia im możliwości interakcji. Lista obsługiwanych tzw. transportów – komponentów odpowiedzialnych za przenoszenie wiadomości pomiędzy aplikacjami, zapewniających zbiór odpowiednich metod specyficznych dla danego kanału transportowego jest dość imponująca i przedstawia się w skrócie następująco: JMS, XMPP, UDP, TCP, SOAP, HTTP, FTP, FILE, EMAIL, SSL, RMI, QUARTZ, JDBC, JBPM, EJB, BPM, MULTICAST, CXF.

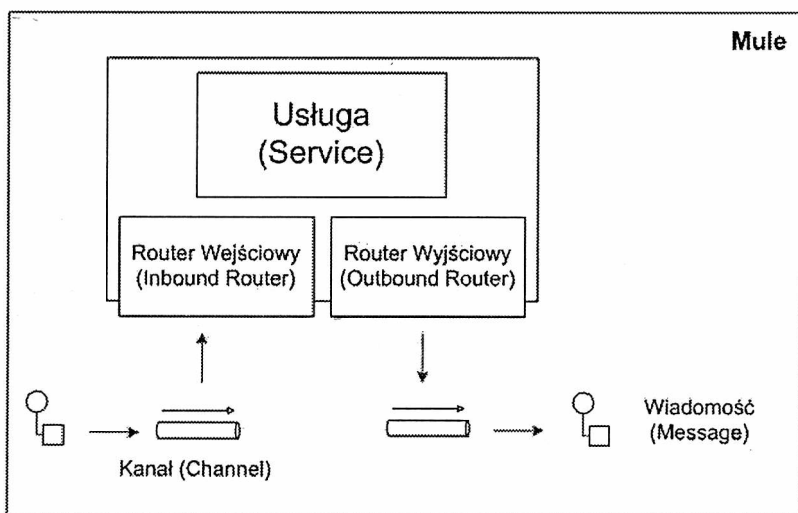
Lista zawiera nazwy własne Mule, ale zawierają one dobrze znane skróty sugerujące wsparcie znacznej części dzisiaj wykorzystywanych technologii. Mule zapewnia ich jednolitą obsługę, oferując przy tym wysoką skalowalność. Zgodnie z zaleceniami SOA szyna zarządza wszystkim interakcjami pomiędzy aplikacjami i komponentami w sposób przezroczysty, niezależnie od użytego systemu operacyjnego, protokołu czy fizycznej lokalizacji.

² Pol. *mul* – który uznawany jest za przewoźnika ładunku, w tym przypadku danych biznesowych.

Przetwarzanie danych w szynie Mule realizowane jest przez komponenty usług osadzone wewnątrz kontenera. Komponenty usług są odseparowane od klas odpowiedzialnych za odbieranie i przesyłanie komunikatów (rys. 4).

Można wyróżnić trzy warstwy oprogramowania tworzonego w środowisku Mule:

- aplikacyjna – składają się na nią wszystkie usługi, które są dostarczane wraz z dystrybucją Mule, pochodzą z zewnętrznych zasobów lub takie, które zostały stworzone dla specyficznych potrzeb;
- integracyjna – w jej skład wchodzi routery, filtry, transformatory oraz pliki konfiguracyjne sterujące przepływem komunikatów;
- transportowa – konektory do obsługi kanałów, transporty oraz inne komponenty wykorzystywane przez środowisko;



Rys. 4. Architektura Mule

Dzięki oparciu o koncepcję szyny ESB, to specyficzne oprogramowanie zapewnia rolę systemu tranzytowego dla wiadomości przekazywanych pomiędzy aplikacjami w intranecie lub Internecie. Kolejną ważną cechą jest neutralność pod względem technologii, z powodu licencji Open Source nie uzależniamy się od konkretnego dostawcy.

3.2. APACHE SERVICEMIX

ServiceMix jest przedstawicielem produktów opartych o specyfikację JBI (ang. *Java Business Integration*). Jest to standard definiujący podejście do implementacji architektury opartej o usługi (SOA) opartej praktycznie w całości na modelu

usług sieciowych. Rozwiązanie to proponuje wykorzystanie kontenera komponentów, który w bardzo prosty sposób pozwala na dołączanie dodatkowych modułów. Usługi łączą się z kontenerem za pośrednictwem specjalizowanych komponentów, a same operacje przeprowadzona są w kolejnych specjalizowanych komponentach.

JBİ obejmuje:

- Środowisko JBİ (ang. *JBİ Enviroment*) – zapewnia wdrażanie komponentów, kontrolę oraz usługi monitorowania. Jest odpowiedzialne za instalowanie modułów i ich cały cykl życia. Odpowiada także za wdrażanie nowych artefaktów dla działających już komponentów.
- Znormalizowany Ruter Wiadomości (ang. *Normalized Message Router, NMR*) – zapewnia infrastrukturę do mediacji wiadomości i jest głównym elementem JBİ. Jest to magistrała, w której wiadomości przesyłane są między nadawcą i odbiorcą w postaci znormalizowanej niezależnie od źródła pochodzenia lub miejsca docelowego (zawsze format XML).
- Komponenty (ang. *Components*) – odgrywają one rolę zarówno dostawców jak i konsumentów usług, dzielą się na:
 - Komponent Usługowy, SE (ang. *Service Engine, SE*) – zapewnia logikę biznesową, usługi transformacji, korzysta także z innych komponentów,
 - Komponent Łączący (ang. *Binding Component, BC*) – zapewnia możliwość komunikacji z usługami zewnętrznymi dla JBİ lub jest to element udostępniający wewnętrzną funkcjonalność.

Różnice między SE a BC są czysto pragmatyczne, ale separacja logiki biznesowej od komunikacyjnej redukuje złożoność i polepsza elastyczność zastosowań (łatwość ponownego wykorzystania).

- Wiadomość Znormalizowana (ang. *Normalized Message*) – jest jedyną jednostką mediacyjną w środowisku JBİ, składa się z dwóch części:
 - właściwej treści w postaci XML,
 - metadanych wykorzystywanych przez NMR i komponenty.

Przedstawione elementy specyfikacji JBİ przedstawione są na rysunku 5. Linia przerywaną został zaznaczony przykładowy przepływ wiadomości pomiędzy komponentami.

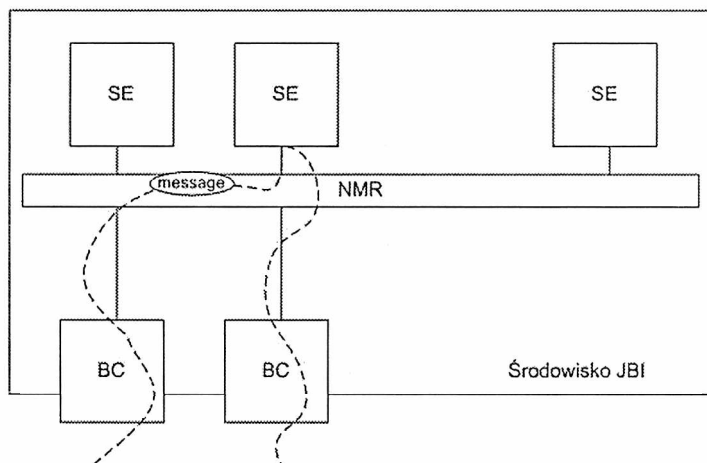
ServiceMix może być uruchamiany jako samodzielny proces lub osadzany w innych kontenerach. Dzięki oparciu o standard pozwala na użycie komponentów pochodzących z innych środowisk JBİ, a jego komponenty mogą być także wykorzystywane w innych implementacjach tej specyfikacji.

Przykładowe moduły dostarczane z podstawową dystrybucją Apache ServiceMix to:

- SE: bean, eip, audit, monitoring, transform, rules,
- BC: EJB, POJO, CXF, JMS, HTTP, EMAIL, JABBER, RSS.

Wykorzystanie komponentów w ServiceMix jest definiowane w plikach xml zawierających dane dotyczące konfiguracji komponentów. Konfiguracja komponentu

jest zależna od implementacji, ale sposób osadzania w kontenerze określa specyfikacja JBI.



Rys. 5. Architektura JBI

4. ZASTOSOWANIA ESB

Wykorzystanie szyn ESB pozwala na realizację wielu różnych scenariuszy. Od najprostszych zapewniających choćby logowanie wykorzystywania danej usługi, po choreografie skomplikowanych skorelowanych procesów biznesowych. Wszystkie je można reprezentować poprzez odpowiednie skonfigurowanie szyny. W literaturze znaleźć można także propozycje wzorców wykorzystywanych w kompozycji scenariuszy integracji [Hohpe 2003].

4.1. WZORCE INTEGRACJI

Korzystanie z wzorców ma wiele dobrze znanych zalet także kiedy mówimy o integracji systemów. Wzorce to szeroko akceptowane rozwiązania dla częstych problemów istniejących w danym kontekście. Są na tyle abstrakcyjne, że pozwalają na wykorzystanie większości technologii, ale na tyle specyficzne, aby zapewnić szczegółowe wskazówki dla programistów i architektów.

Wzorce integracyjne składają się z małych elementów, które pozwalają budować skomplikowane struktury integracyjne. Takimi elementami są np.:

Takie elementy dają się dość łatwo aranżować i stosować przy modelowaniu procesów, przykładowy proces może przebiegać następująco:

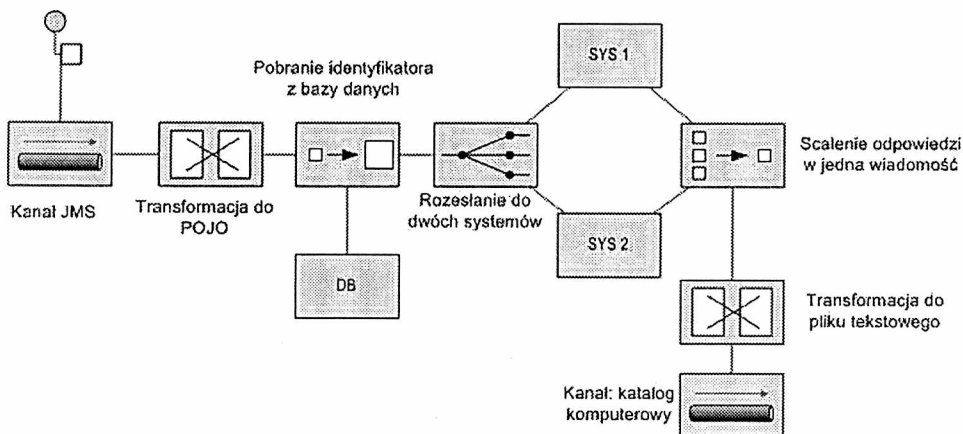
Proces biznesowy realizuje funkcjonalność wyznaczania wartości rachunku klienta za usługi telekomunikacyjne. Proces przebiega następująco:

- 1. Wysłanie żądania w postaci wiadomości JMS, zawierającej dane osobowe klienta.
- 2. Transformacja wiadomości do obiektu POJO(ang. *Plain Old Java Object*).
- 3. Wzbogacenie wiadomości o identyfikatora klienta, pobrany z bazy danych.
- 4. Wysłanie zapytania o koszty do systemu odpowiedzialnego za rozmowy lokalne oraz do systemu odpowiedzialnego za rozmowy zamiejscowe.
- 5. Scalenie wiadomości w jedną.
- 6. Transformacja do odpowiedniej postaci (format obsługiwany przez drukarki, np. tekst).
- 7. Skierowanie do odpowiedniego katalogu komputerowego, z które zostanie pobrany plik tekstowy i np. wydrukowany.

Tabela 1. Kilka elementów wzorców integracji

	Wiadomość – jednostka wymiany informacji w przedsiębiorstwie
	Kanał – medium komunikacyjne pomiędzy aplikacjami
	Agregator – scala kilka wiadomości w jedną
	Wzbogacz wiadomości – element dodający pewne dane do już istniejących
	Router trasujący wiadomości po typie zawartości
	Lista odbiorców danej wiadomości
	Transformator wiadomości (np. protokołu lub formatu)
	Rozdzielacz – przeciwieństwo agregatora

Na rysunku 6 przedstawiono sposób, w jaki można zamodelować proces za pomocą elementów wzorców. Dla uproszczenia kilka elementów zostało pominiętych, jak na przykład transformacja formatu przy komunikacji z bazą danych lub z systemami bilingowymi.

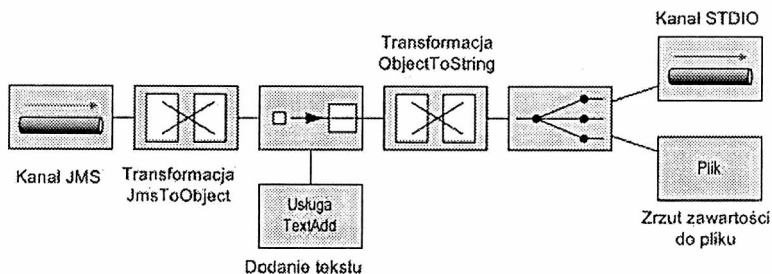


Rys. 6. Przykładowy proces biznesowy opisany za pomocą elementów wzorców integracji

4.2. PRZYKŁAD – MULE

Konfiguracja Mule sprowadza się do odpowiedniego wyedytowania stosownych plików .xml oraz ewentualnego oprogramowania specyficznych elementów systemów np. nieobsługiwanego standardowo protokołu lub transformacji. Sytuacja taka zdarza się rzadko, ponieważ Mule posiada pokaźne zbiory gotowych elementów do wykorzystania.

Dla przykładu przeprowadzono konfigurację Mule do obsługi poniżej przedstawionego prostego procesu.



Rys. 7. Prosty przykład procesu implementowany w Mule

Jak pokazano na rysunku 7, mamy do czynienia z wysłaniem wiadomości tekstowej za pomocą wiadomości JMS, która następnie transformowana jest do obiektu Java – zwykłego Stringa, wzbogacana o dowolny tekst i na koniec rozesłana w dwa miejsca – konsole monitora (kanał STDOUT) oraz plik.

Konfiguracja przedstawia się następująco:

```
<jms:activemq-connector name="jmsCon" brokerURL="tcp://localhost:61616" />
<jms:jmsmessage-to-object-transformer name="JmsToObject" />
<append-string-transformer name="myAppender" message=" dodatkowyTekst" />
<object-to-string-transformer name="ObjectToString" />

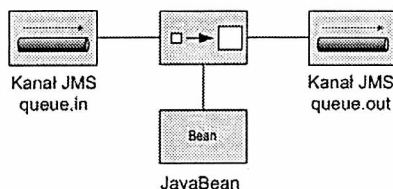
<model name="SampleModel">
  <service name="ServiceSample">
    <inbound>
      <jms:inbound-endpoint queue="queue.in"
        transformer-refs="JmsToObject ObjectToString myAppender" />
    </inbound>

    <outbound>
      <multicasting-router>
        <file:outbound-endpoint path="LogPath"
          outputPattern="${DATE:HH-mm-ss}-logger.log" />
        <stdio:outbound-endpoint system="OUT" />
      </multicasting-router>
    </outbound>
  </service>
</model>
```

Pominięto tu niektóre elementy jak deklaracja przestrzeni nazw. Cały proces reprezentowany jest przez model o nazwie „SampleModel”, które zawiera jedną usługę o nazwie „ServiceSample”. Jest on złożony z dwóch punktów wejściowego (ang. *inbound*) oraz wyjściowego (ang. *outbound*). Punkt wejściowy odbiera wiadomości JMS z kolejki o nazwie *queue.in* po czym transformuje kolejno do obiektu, Stringu, dodając na końcu dodatkowy tekst. Punkt wyjściowy składa się routera rozsyłającego kopie tych samych elementów do wyspecyfikowanych miejsc, w naszym przypadku jest to plik oraz konsola wyjściowa na monitorze.

4.3. PRZYKŁAD W SERVICEMIX

Przykład prezentowany dla ServiceMix’a przedstawiono na rysunku 8 [WW2]:



Rys. 8. Prosty przykład procesu implementowanego w ServiceMix

Przepływ prezentowany na rysunku 8 jest podobny do tego prezentowanego dla Mule, z takim wyjątkiem, że do wzbogacenia wiadomości będziemy używać klasy

Javy, która doda tekst do wiadomości, po czym skieruje ją do kolejki wyjściowej także w postaci wiadomości JMS.

```
<beans>
<bean id="connectionFactory" >
  <property name="brokerURL" value="tcp://localhost:61616"/>
</bean>

<jms:consumer service="esb:simpleReceiver"
  endpoint="jmsEndpoint"
  targetService="esb:helloComponent"
  destinationName="in.queue"
  replyDestinationName="out.queue"
  connectionFactory="#connectionFactory" />

<jsr181:endpoint annotations="none" service="esb:helloComponent"
endpoint="endpoint" serviceInterface="esb.example.HelloIF" id="hello">
  <jsr181:pojo>
    <bean id="helloBean" class="esb.example.HelloBean">
      <property name="prefix" value="Hello" />
    </bean>
  </jsr181:pojo>
</jsr181:endpoint>

<eip:pipeline service="esb:examplePipeline" endpoint="routingEndpoint">
  <eip:transformer>
    <eip:exchange-target service="esb:helloComponent" />
  </eip:transformer>
  <eip:target>
    <eip:exchange-target service="esb:simpleSender" />
  </eip:target>
</eip:pipeline>
</beans>
```

W przypadku konfiguracji w ServiceMixie należało stworzyć prostą klasę Javy która dodawała odpowiedni tekst. Poza tym kod xml jest w miarę intuicyjny. Widać wykorzystanie takich komponentów jak JMS (jako BC), EIP (do trasowania wiadomości), oraz JSR181 (do wykorzystywania klas Javy).

5. PODSUMOWANIE

Korporacyjne szyny usług zdobywają w ostatnim czasie coraz więcej zastosowań. Dzięki swoim funkcjom rozwiązują szereg problemów integracyjnych w dzisiejszych korporacjach. Ich zastosowanie pozwala na łatwiejsze wprowadzanie modyfikacji w przyszłości. Dzięki oparciu o otwarte standardy unikamy uzależnienia od konkretnego dostawcy i tym samym umożliwiamy sobie wybór najlepszych rozwiązań.

Na rynku dostępnych jest kilkanaście implementacji szyn ESB, także specjalizowanych dla różnych rozwiązań biznesowych. Dwie zaprezentowane w pracy propozycje dostępne są na zasadzie open source i są obecnie dynamicznie rozwijane.

LITERATURA

- [Binil 2008] Binildas C.A., *Service Oriented Java Business Integration*, Packt Publishing, 2008.
- [Chapp 2004] Chappel D., *Enterprise Service Bus*, O'Reilly, 2004.
- [Dirk 2008] Dirksen J., Rademakers T., *Open-Source ESBs in Action*, Manning Publications, 2008.
- [Hohpe 2003] Hohpe G., Woolf B., *Enterprise Integration Patterns*, Addison-Wesley, 2003
- [Linh 2000] Linthicum D., *Enterprise Application Integration*, Addison-Wesley, 2000.
- [WWW1] http://www.ebizq.net/topics/int_sbp/features/3463.html
- [WWW2] <http://architects.dzone.com/articles/pattern-based-development-with-0?page=0%2C1>.

ENTERPRISE SERVICE BUSES

This article presents a brief overview of enterprise service buses (ESB) with the key aspects of their usage in application integration. An introduction to application integration using service oriented architecture is presented. Expected functions of ESB are shown. Finally two open-source ESB implementations are described: Mule and ServiceMix. Examples of simple business processes implementations using both products are included.

*podpis elektroniczny,
repozytorium certyfikatów,
szyfrowanie i podpisywanie dokumentów*

Damian CZERNOUS*

PODPIS ELEKTRONICZNY ZNAKIEM NOWEJ ERY

Niniejszy artykuł jest poświęcony zagadnieniom podpisu elektronicznego. Poruszono w nim aspekt technologiczny, prawny, a także psychologiczny zagadnienia. Dlatego odbiorcą może być każda osoba ciekawa tematu, niezależnie od swojej specjalizacji. Na szczególną uwagę zasługuje rozdział „Podpis elektroniczny”, który wyjaśnia, na czym polega podpisanie dokumentu elektronicznego. Natomiast programistów powinien zainteresować fragment dotyczący obsługi repozytorium certyfikatów składowanych zarówno na karcie kryptograficznej jak i w pliku (język kodowania: Java). Całość, niejednokrotnie prezentująca śmiałe wnioski, jest bogato ilustrowana oraz oparta na przykładach.

1. WPROWADZENIE

Studiowanie duszy (greckie: „psyche” i „logos”) jest dosłownym tłumaczeniem słowa psychologia. Psychologia to dziedzina nauki zajmująca się badaniem zachowań i procesów umysłowych ludzi. Już przeszło 2400 lat temu grecki lekarz Hipokrates podzielił ludzi, ze względu na ich zachowania, na cztery typy temperamentalne: sangwinik, choletryk, melancholik i flegmatyk. Człowiek na ogół jednak nie reprezentuje w całości jednego typu, ale raczej ich kombinację. W wyniku tej mieszaniny, przyjmujemy odmienne postawy w napotkanych sytuacjach. Obok łatwych do interpretacji wyraźnych reakcji, istnieją również znacznie bardziej subtelne, dostarczające ciekawych informacji o nas samych. Może to być np. sposób w jaki się poruszamy, stoimy, patrzymy lub piszemy.

1.1. SZTUKA SŁOWA PISANEGO

Pismo odręczne, to jeden z bardziej wdzięcznych sposobów komunikacji, które nadało myślom nieśmiertelność, uwiecznia historię ludzkości, zapewnia rozrywkę,

* Instytut Informatyki, Wydział Informatyki i Zarządzania Politechniki Wrocławskiej, 50-370 Wrocław, Wybrzeże Wyspiańskiego 27, damian.czernous@pwr.wroc.pl

a także pozwala zanotować codzienną listę zakupów. Niemiecki złotnik i drukarz Jan Gutenberg (1399–1468), który stworzył pierwsze w Europie duże wydawnictwo (1448 r.), wykorzystał potencjał pisma drukowanego, zapewniając jednocześnie wielu ludziom, łatwy i tani dostęp do książki. Był to wyraźny skok w przyszłość.

Czterysta lat później (w 1883 roku) Charles Babbage zaprojektował maszynę analityczną uznawaną do dziś za pierwszy komputer. Tak, w dużym uproszczeniu, minęły całe stulecia, aż po czasy współczesne. Paradoksalnie dzisiejszy człowiek niezmiernie rzadko używa pióra, a te nieliczne przypadki, najczęściej ograniczają się do podpisywania różnego rodzaju dokumentów. Zatem dlaczego w erze pisma drukowanego, a wręcz cyfrowego wymagamy odręcznego podpisu (niejako archaizmu) pod dokumentem, który ma przedstawiać jakąś wartość?

Cała odpowiedzialność spoczywa w rękach osobowościowego indywidualizmu, który czyni z nas osoby niepowtarzalne. Zatem wszystko co robimy, również musi takim być, włączając pismo odręczne. Odwracając sytuację – po sposobie pisania, można określić cechy osobowościowe pisarza. Być może dlatego grafologia jest przedmiotem psychologii. Reasumując, indywidualność psychiczna determinuje niepowtarzalność podpisu. W ten łatwy sposób, odpowiadamy na tak postawione pytanie. Wśród zajęć grafologa, odnaleźć możemy umiejętność udowodnienia autentyczności podpisu złożonego pod dokumentem. Zatem zawód ten, jest zajęciem z przeszłością. Ale czy na pewno?

1.2. PODPIS ELEKTRONICZNY – KOLEJNYM KROKIEM W PRZYSZŁOŚĆ

Podpis elektroniczny, w odróżnieniu od formy „analogowej” (tej odręcznej), ma postać wyłącznie cyfrową. Mimo to i tak zapewnia, jego właścicielowi, niepowtarzalność, a także możliwość weryfikacji jego autentyczności. Jest bardzo ważnym, aby cechy unikalności i autentykacji formy analogowej były w pełni zachowane. Na nic tu jednak się zdadzą umiejętności grafologa, gdyż cechy te nie mają podłoża osobowościowego, a matematyczne. Jeszcze wiele lat upłynie, zanim grafologia będzie zawodem na wyginieciu. Jednak w dobie informacji cyfrowej, transakcji elektronicznej i Internetu wydaje się nie mieć pewnej przyszłości.

W przypadku podpisu elektronicznego rolę grafologa pełni Urząd Certyfikacji (ang. *Certification Authority* – CA). W Polsce istnieją trzy Urzędy Certyfikacji: Siggillum PCCE (Polskiej Wytwórni Papierów Wartościowych – PWPW S.A.), KIR (Krajowej Izby Rozliczeniowej) oraz Unicert (Unizeto S.A.). Należy zauważyć, że technologia ta nie jest nowa. Już od dawna jest szeroko wykorzystywana (np. certyfikaty elektroniczne używane w połączeniach SSL). Jednak w obliczu niesprawnego prawodawstwa, i co więcej – braku odpowiednich aktów prawnych, jest wolną fantazją sympatyków i zwolenników tego typu rozwiązań. Podstawą do funkcjonowania

podpisu jest Ustawa o podpisie elektronicznym z dnia 18 września 2001 roku wraz z aktami wykonawczymi. Definiuje ona dwa rodzaje podpisu: kwalifikowany i zwykły. Obecnie przygotowywana jest nowelizacja ustawy wprowadzająca trzecią postać – uniwersalną. Analiza regulacji prawnych jest tematem na osobny artykuł. Warto jednak wspomnieć o czynniku ludzkim w kwestii popularyzacji tej technologii. Bowiem wszystko co nowe i nieznane spotyka się z naturalnym uprzedzeniem i ostrożnością potencjalnego użytkownika. Dobrym tutaj przykładem jest pojawienie się karty bankomatowej, która w pierwszej chwili zetknęła się z szeroką krytyką. Najcięższym zarzutem była kwestia bezpieczeństwa. Mimo, że typowy użytkownik dalej nie rozumie, dlaczego bezpieczne jest używanie takiej karty, to jednak ją w pełni akceptuje i można śmiało powiedzieć, że stała się nieodzownym elementem jego życia. Odpowiedzialność za ten stan rzeczy leży po stronie reklamy oraz oczywistych korzyści, jakie daje używanie karty. Inaczej mówiąc, wygoda wzięła górę nad wątpliwościami. A ponieważ reklama przekonuje, że jest to bezpieczne urządzenie, tym szybciej zyskała na popularności. Podobnie jest z podpisem elektronicznym, z tą jednak różnicą, że brakuje reklamy i sprawnej infrastruktury informatycznej, mogącej wykorzystać potencjał jaki w nim drzemie.

Jak można zauważyć, świat cyfrowy stara się naśladować ten realny. W ten sposób jest łatwy w akceptacji i bardziej zrozumiały. W dalszej części artykułu omówiono Infrastrukturę Klucza Publicznego (PKI), wyjaśniono różnice i przeznaczenie wymienionych dwóch typów podpisu elektronicznego, objaśniono na czym polega szyfrowanie, deszyfrowanie, podpisywanie i weryfikacja podpisu, przedstawiono sposób generowania podpisu i związanych z nim certyfikatów. Na koniec pokażemy, jak uzyskany w ten sposób podpis można wykorzystać podczas redagowania wiadomości elektronicznej.

2. INFRASTRUKTURA KLUCZA PUBLICZNEGO (PKI)

Idea PKI oparta jest na cyfrowych certyfikatach potwierdzających związek między określonymi uczestnikami transakcji a kluczami kryptograficznymi, użytymi podczas transakcji. Certyfikat cyfrowy jest wydawany przez Urząd Certyfikacji, który w momencie wydania certyfikatu potwierdza własnym podpisem elektronicznym związek pomiędzy użytkownikiem, a kluczem, którego używa. Ponieważ tak wystawiony certyfikat jest zawsze ograniczony okresem ważności, przewiduje się następujące sytuacje związane z jego wykorzystaniem: rejestracja użytkownika, generowanie certyfikatów, dystrybucja, aktualizacja i unieważnianie. Z tego właśnie powodu, struktura PKI oparta została na czterech głównych elementach:

- Urząd Rejestracji (ang. *Registration Authority* – RA), który zajmuje się weryfikacją danych użytkownika, a następnie jego rejestracją.
- Urząd Certyfikacji (ang. *Certification Authority* – CA), wydającym certyfikaty cyfrowe.

- Repozytoriach kluczy, certyfikatów i list unieważnionych certyfikatów (ang. *Certificate Revocation Lists* – CRL), które umożliwiają, w oparciu o protokół LDAP, dostęp do certyfikatów i list CRL. Inne sposoby realizacji dostępu mogą być oparte na protokołach X.500, HTTP, FTP, a także poczcie elektronicznej. Certyfikat może zostać unieważniony w trakcie swojego naturalnego okresu ważności. Przyczyną takiego zjawiska jest np. zmiana nazwiska lub adresu poczty elektronicznej użytkownika (w przypadku, gdy został wpisany do certyfikatu) lub ujawnienie (skompromitowanie) klucza prywatnego. W takich sytuacjach CA unieważnia certyfikat poprzez umieszczenie numeru seryjnego na ogólnodostępnej liście CRL. Obecnie powstała nowa możliwość weryfikacji certyfikatów za pośrednictwem usługi OCSP. Potrzeba taka wynika z wady list CRL, jaką jest ich okresowa aktualizacja. Odbывается ona raz na godzinę. Natomiast w przypadku OCSP otrzymujemy natychmiast pewną informację na temat statusu całego certyfikatu.
- Użytkownikach końcowych (ang. *End Entity*) – czyli osobach, aplikacjach, sprzęcie.

Struktura PKI jest tworzona w oparciu o główne CA, przy czym dla każdego z obszarów zastosowań (np. handel elektroniczny, sektor bankowo – finansowy, administracja publiczna), które będą korzystać z PKI, można tworzyć odrębne CA naturalnie podległe głównemu urzędowi. To główne CA (ang. *root CA*) określa ogólną politykę certyfikacji, przy czym podległe CA, odpowiada za utrzymanie tej polityki w ramach przypisanego obszaru zastosowań. Dopiero z podległym CA związany jest użytkownik końcowy tzw. subskrybent. Jak łatwo zauważyć taka struktura tworzy hierarchię uwierzytelniania, wiodącą od użytkowników, aż do głównego CA (jest to tzn. ścieżka certyfikacji). Krajowa struktura PKI musi współdziałać ze strukturami PKI innych krajów, aby zapewnić usługi o podobnym charakterze w kontaktach międzypaństwowych. Obecnie polska struktura PKI odróżnia się od struktur naszych sąsiadów. Zmienić ten stan rzeczy ma wciąż przygotowywana nowelizacja Ustawy o podpisie elektronicznym.

2.1. PODSTAWOWE FUNKCJE PKI

Opiszemy teraz obligatoryjne funkcje PKI, zapewniające pożądaną poziom usług.

Rejestracja (ang. *registration*). Subskrybent końcowy składa wniosek do organu Rejestracji z prośbą o wydanie certyfikatu. Jest to związane z dostarczeniem szeregu informacji, wymaganych przez Kodeks Postępowania Certyfikacyjnego (ang. *Certification Practices Statement* – CPS) danego CA np. nazwa własna, nazwa domenowa czy adres IP. Zanim CA wystawi certyfikat, sprawdza, czy podane przez użytkownika dane są zgodne z prawdą. Często praktyką jest konieczność osobistego stawiennictwa przyszłego subskrybenta w RA. Jeżeli o certyfikat ubiega się osoba fizyczna, CA weryfikuje także autentyczność własnoręcznego podpisu na wniosku o wydanie certyfi-

katu. Zanim posłużymy się formą cyfrową podpisu raz jeszcze należy ją potwierdzić tradycyjnym podpisem odręcznym. Jest to element wiążący te dwa światy (realny i cyfrowy).

Certyfikacja (ang. *certification*). Jeżeli weryfikacja przyszłego subskrybenta przebiegnie poprawnie, CA wystawia nowy certyfikat (zawierający między innymi klucz publiczny). Jednocześnie certyfikat zostaje umieszczony we właściwym repozytorium publicznym, ale tylko wtedy, gdy jego właściciel wyrazi na to zgodę. Z punktu widzenia użyteczności takiego certyfikatu, nie umieszczenie go w publicznym repozytorium skutkuje trudnościami z jego weryfikacją przez różnorodne systemy informatyczne. Jest to zupełnie naturalna sytuacja, ponieważ nie certyfikat mamy chronić, lecz klucz prywatny. Jak to dokładnie wygląda przedstawiono w dalszej części artykułu.

Generacja kluczy (ang. *key generation*). Para kluczy (prywatny i publiczny) może zostać wygenerowana samodzielnie przez użytkownika końcowego lub przez CA. Zwyczajowo wykonuje za nas tę czynność CA, ponieważ jest to łatwiejsze. W pierwszym przypadku użytkownik przesyła do CA tylko klucz publiczny w celu poddania go procesowi certyfikacji. Klucz prywatny pozostaje przez cały czas w rękach właściciela, dlatego też metodę tę uważa się za najbardziej bezpieczną. Jeżeli generowaniem kluczy zajmuje się CA, to są one dostarczane do użytkownika końcowego w sposób gwarantujący ich poufność. Najchętniej wykorzystuje się do tego celu karty mikroprocesorowe (ang. *SmartCard*) lub karty PCMCIA (ang. *Personal Computer Memory Card International Association*), obie zabezpieczone dodatkowym kodem PIN. W przypadku ubiegania się o certyfikat kwalifikowany użytkownik nie może samodzielnie wygenerować pary kluczy ponieważ polskie CA, zgodnie z przyjętą polityką, w tym przypadku uznają zasadę istnienia takiego klucza wyłącznie na karcie kryptograficznej.

Odnawianie kluczy (ang. *key update*). Wszystkie pary kluczy oraz powiązane z nimi certyfikaty są okresowego odnawiane. Jest to kolejne zabezpieczenie na wypadek ujawnienia klucza prywatnego skojarzonego z kluczem publicznym umieszczonym w certyfikacie. Można mówić o dwóch przypadkach, w których zachodzi konieczność wymiany kluczy:

- wygaśnięcie certyfikatu (jest to sytuacja normalna),
- skompromitowanie klucza prywatnego (jest to sytuacja wyjątkowa). CA zamieszcza taki certyfikat na wspomnianej już liście CRL (*Certificate Revocation List*) i rozpoczyna procedurę ponownego wystawiania certyfikatu. Najgorszy przypadek, dla każdego CA, to kompromitacja klucza prywatnego głównego CA. W takim przypadku cała infrastruktura uznawana jest za skompromitowaną i wszystkie certyfikaty z nią związane stają się nieważne.

Certyfikacja wzajemna (ang. *cross – cerification*). Społeczność międzynarodowa nie utworzyła dotychczas Globalnego Organu Certyfikacji (ang. *Global Root CA*). Powstało wiele Głównych Organów Certyfikacji nie powiązanych wzajemnymi relacjami zaufania. Certyfikacja wzajemna pozwala subskrybentom jednej struktury PKI

ufać subskrybentom innej. Główne CA z różnych struktur certyfikują się wzajemnie (certyfikacja jednokierunkowa lub dwukierunkowa).

Odwołanie certyfikatu (ang. *revocation*). Jest kilka powodów, dla których zachodzi konieczności odwołania certyfikatu: kompromitacja klucza prywatnego, zmiana nazwy użytkownika końcowego, odejście pracownika z firmy, która wystawiła mu certyfikat. Określona w standardzie X.509 metoda odwoływania certyfikatu wykorzystuje właśnie listy CRL (dla przykładu, Sigillum PCCE aktualizuje listy co 1 godzinę). Każdy certyfikat posiada swój unikalny numer seryjny przypisany przez CA w momencie jego wystawiania.

Odzyskiwanie klucza (ang. *key recovery*). To dodatkowe zabezpieczenie pozwala użytkownikowi odzyskać swoje klucze. Jeżeli wszystkie klucze do szyfrowania były przechowywane w bezpiecznym archiwum, to można je odzyskać. Przy czym najważniejsza jest gwarancja, że klucze będzie mógł odzyskać tylko ich właściciel, a nie osoba trzecia.

2.2. ZASTOSOWANIE PKI

Można wyróżnić następujące usługi uwierzytelniania:

- uwierzytelnianie podmiotów partnerskich (ang. *peer – entity authentication*), pozwalające na pewną identyfikację przez innych użytkowników,
- uwierzytelnianie danych (ang. *data authentication*), pozwalające stwierdzić, że informacja była świadomie podpisana przez danego użytkownika,
- uwierzytelnianie integralności danych (ang. *data integrity*), określające, czy informacja podpisana cyfrowo nie została zmieniona,
- usługę niezaprzeczalności (ang. *non – repudation*), uniemożliwiająca użytkownikom późniejsze wyparcie się faktu użycia podpisu (grafolog). W Polsce jest to cecha certyfikatu kwalifikowanego. W praktyce odpowiednie ustawienie ostatniego bitu w certyfikacie (wartość 0 lub 1),
- usługi poufności:
 - poufność (ang. *confidentiality*), umożliwiająca użytkownikom ochronę danych przed nieuprawnionym ujawnieniem,
 - prywatność (ang. *privacy*), pozwalająca użytkownikom na wymuszenie specjalnej obsługi informacji podczas jej transmisji w centrach łączności, pomiędzy centrami łączności, a także odbiorcami końcowymi. Dotyczy to również przechowywania informacji.

W związku z powyższym aplikacje korzystające z PKI powinny umożliwić:

- szyfrowanie danych dla zapewnienia ich poufności,
- cyfrowe podpisanie danych dla zapewnienia niezaprzeczalności i weryfikacji ich integralności,
- uwierzytelnienie osób, aplikacji, serwisów.

Obszary zastosowań podpisu elektronicznego:

- bezpieczna poczta elektroniczna,
- transakcja elektroniczna (np. aukcja, licytacja, zakupy elektroniczne),
- wirtualna sieć prywatna (ang. *Virtual Private Network* – VPN),
- systemy ERP (zarządzania zasobami przedsiębiorstw),
- zabezpieczenie stacji roboczej użytkownika,
- zapewnienie bezpieczeństwa w witrynach internetowych, urządzeniach i aplikacjach klienta.

2.3. STANDARDY PKI

PKI zasadniczo można podzielić na dwie grupy: standardy poziomu użytkownika i standardy samego PKI. Zastosowanie określonych standardów w strukturze PKI pozwala na współpracę między różnymi jej strukturami i powtarzalne wykorzystanie tej infrastruktury w wielu aplikacjach. Standaryzacja obejmuje swoim zasięgiem:

- procedury rejestracji,
- formaty certyfikatów,
- formaty list CRL,
- formaty podpisów cyfrowych,
- protokoły typu *challenge/response*.

Próbie standaryzacji funkcji PKI podjęła swego czasu grupa robocza IETF (ang. *Internet Engineering Task Force*), znana również jako grupa PKIX (PKI dla certyfikatów X.509). Cztery podstawowe składniki modelu PKIX to: subskrybent, CA, RA, a także repozytorium certyfikatów. Specyfikacja PKIX opiera się na dwóch standardach: X.509 Międzynarodowego Związku Telekomunikacji (ang. *International Telecommunication Union* – ITU) i PKCS Standardów Kryptografii Klucza Publicznego (ang. *Public Key Cryptography Standards*) autorstwa *RSA Data Security*. Najpopularniejsze standardy PKCS to:

- PKCS#7 – standard kryptograficznego kodowania wiadomości (ang. *Cryptographic Message Syntax Standard*),
- PKCS#10 – standard kodowania wniosku o certyfikat (ang. *Certificate Request Syntax Standard*),
- PKCS#12 – standard kodowania informacji poufnych w postaci plików (ang. *Personal Information Exchange Syntax Standard*).

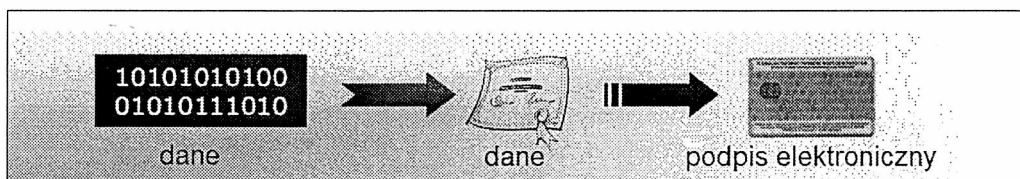
Należy zauważyć, że większość najważniejszych standardów w dziedzinie bezpieczeństwa teleinformatycznego jest tak pomyślana, aby zabezpieczyć współpracę z PKI. Dobrymi tu przykładami są:

- SSL (ang. *Secure Socket Layer*),
- TLS, SMIME (ang. *Secure Multipurpose Internet Mail Extensions*),
- SET (ang. *Secure Eletronic Transactions*), IPSec (ang. *IP Security*).

3. PODPIS ELEKTRONICZNY

Według [Ustawa 2001] podpis elektroniczny, to „dane w postaci elektronicznej, które wraz z innymi danymi, do których zostały dołączone lub z którymi są logicznie powiązane, służą do identyfikacji osoby składającej podpis elektroniczny”.

Niejasność tej definicji objaśniono na rysunku 1. Otóż, dane w postaci elektronicznej (innymi mówiąc para kluczy), które wraz z innymi danymi (certyfikat elektroniczny podpisany przez zaufany Urząd Certyfikacji), do których zostały dołączone lub z którymi są logicznie powiązane (certyfikat jest ściśle powiązany z parą kluczy), służą do identyfikacji osoby składającej podpis elektroniczny.



Rys. 1. Definicja podpisu elektronicznego

Kolejna definicja dotyczy bezpiecznego podpisu elektronicznego.

„Bezpieczny podpis elektroniczny – podpis elektroniczny, który:

Jest przyporządkowany wyłącznie do osoby składającej ten podpis.

Jest sporządzany za pomocą podlegających wyłącznej kontroli osoby składającej podpis elektroniczny bezpiecznych urządzeń służących do składania podpisu elektronicznego i danych służących do składania podpisu elektronicznego. Jest powiązany z danymi, do których został dołączony, w taki sposób, że jakakolwiek późniejsza zmiana tych danych jest rozpoznawalna” [Ustawa 2001].

Każdy podpis elektroniczny jest związany z jedną konkretną osobą, która przy pomocy bezpiecznego urządzenia podpisuje dokument elektroniczny. Taką sytuację zilustrowano na rysunku 2. Bezpieczny podpis elektroniczny to taki, który umożliwi wykrycie jakichkolwiek zmian w dokumencie wcześniej podpisanym.

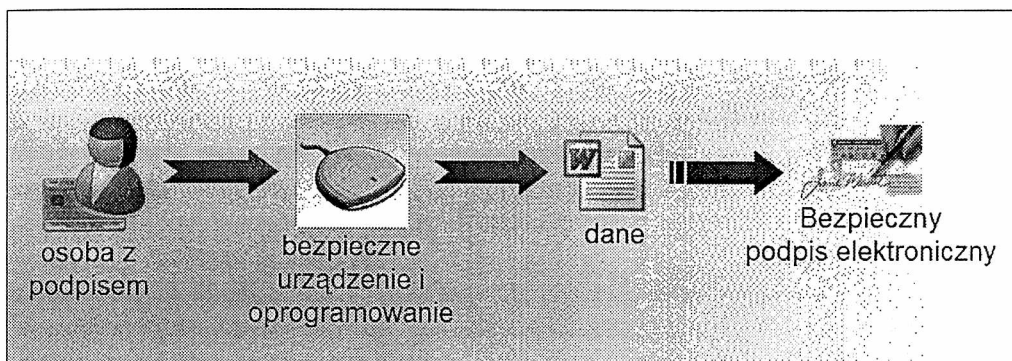
Podobieństwa podpisu elektronicznego z podpisem własnoręcznym:

- służy do identyfikacji osoby,
- stanowi dowód złożenia oświadczenia woli.

Cechy podpisu elektronicznego:

- Podpis elektroniczny nie ma postaci graficznej,
- Nie istnieje w oderwaniu od podpisywanych danych,
- Jest dynamiczny,
- Ma czasową skuteczności (1 lub 2 lata),
- Nie odzwierciedla cech fizycznych osoby,

- Jest składany za pomocą bezpiecznych urządzeń,
- Można posiadać więcej niż jeden podpis elektroniczny.



Rys. 2. Definicja bezpiecznego podpisu elektronicznego

Jak już wspomniano, w Polsce funkcjonują dwa rodzaje podpisu elektronicznego:

- kwalifikowany podpis elektroniczny,
- zwykły podpis elektroniczny nazywany też komercyjnym.

Kwalifikowany podpis elektroniczny definiowany jest następująco:

„Dane w postaci elektronicznej opatrzone bezpiecznym podpisem elektronicznym weryfikowanym przy pomocy ważnego kwalifikowanego certyfikatu są równoważne pod względem skutków prawnych dokumentom opatrzonym podpisami własnoręcznymi, chyba że przepisy odrębne stanowią inaczej” [Ustawa 2001].

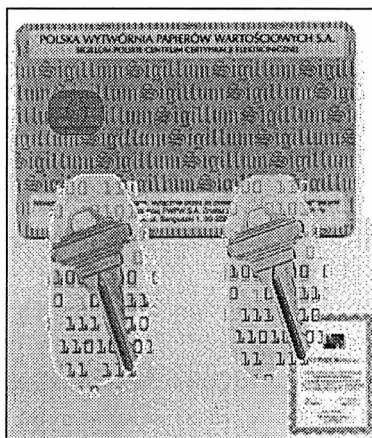
Może on znaleźć zastosowanie również w:

- pismach procesowych,
- podaniach administracyjnych,
- orzeczeniach sądowych i decyzjach administracyjnych,
- wezwaniach w zakresie procedury administracyjnej i cywilnej,
- fakturach.

W odróżnieniu od komercyjnego podpisu, który można użyć w systemach teleinformatycznych do weryfikacji tożsamości podpisywania dokumentów takich jak poczta elektroniczna lub szyfrowania danych, dokument opatrzone podpisem kwalifikowanym zyskuje moc prawną (wynika to z zapisów ustawy). Należy jednak zachować ostrożność, ponieważ nie każdy dokument elektroniczny, w rozumieniu informatycznym, jest dokumentem w sensie prawnym. Ponadto ustawa dokładnie precyzuje, które dokumenty mogą być opatrzone takim podpisem. Podpis komercyjny również może mieć taką moc, jeśli dwie osoby (posiadające taki podpis) umówią się co do jego ważności (np. podpiszą odrębne oświadczenie woli i opatrzą je kwalifikowanym podpisem elektronicznym).

3.1. DZIAŁANIE PODPISU ELEKTRONICZNEGO

Klucz prywatny powinien być utrzymywany w sekrecie (jako dane poufne). W przypadku podpisu kwalifikowanego, nie opuszcza on repozytorium karty kryptograficznej, do którego został raz zapisany (przyjęta polityka i własność karty). Zasada ta nie obowiązuje klucza prywatnego podpisu komercyjnego, jako że może występować w wielu repozytoriach (karta, plik, przeglądarka internetowa itp.). Natomiast klucz publiczny może być udostępniony np. w Internecie. Dane podpisuje się za pomocą klucza prywatnego. Wzajemne powiązania pomiędzy kartą kryptograficzną, parą kluczy, a certyfikatem przedstawiono na rysunku 3.



Rys. 3. Kryptografia asymetryczna – para kluczy (prywatny oraz publiczny)

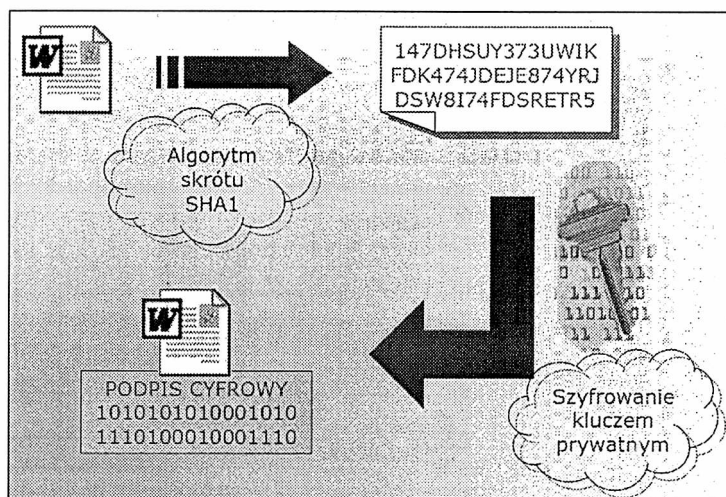
W przypadku szyfrowania zasada jest następująca. Dane zaszyfrowane kluczem prywatnym, mogą być odszyfrowane jedynie kluczem publicznym i na odwrót. Aby proceder ten miał sens, powinno się szyfrować dane kluczem publicznym osoby upoważnionej do zapoznania się z ich treścią. Ten za pomocą klucza prywatnego będzie wstanie je odszyfrować.

3.2. SKŁADANIE PODPISU ELEKTRONICZNEGO

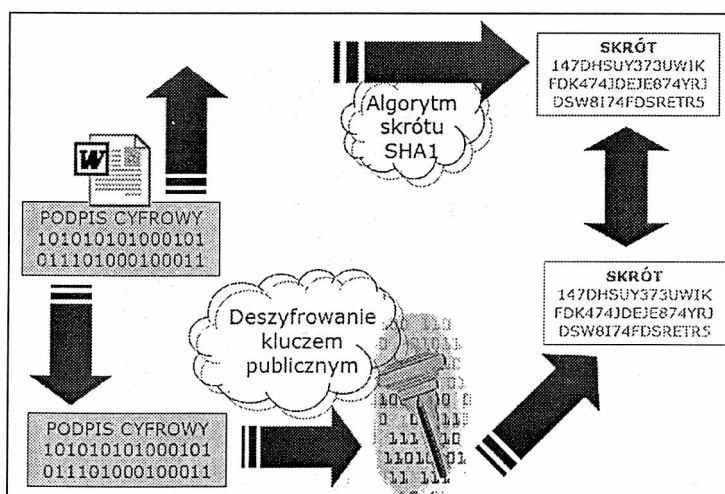
Na rysunkach 4 i 5 przedstawiono kolejne kroki procesu składania i weryfikacji podpisu. Za pomocą funkcji skrótu SHA1 tworzony jest skrót dokumentu, który ma być podpisany. Skrót ten ma długość 160 bitów niezależnie od długości danych wejściowych – tzn. „odcisk palca”. Na podstawie skrótu, nie można odtworzyć dokumentu, ale zmiana 1 bitu dokumentu wejściowego spowoduje zmianę co najmniej połowy bitów skrótu.

Tak utworzony skrót jest szyfrowany za pomocą klucza prywatnego. Całość dołączana jest do dokumentu.

Podczas weryfikacji podpisu ponownie tworzony jest skrót dokumentu. Kluczem publicznym deszyfrowany jest uprzednio spreparowany podpis. W rezultacie, otrzymuje się, stworzony w czasie składania podpisu skrót dokumentu. Oba skróty należy porównać. Zgodność oznacza nienaruszony dokument.



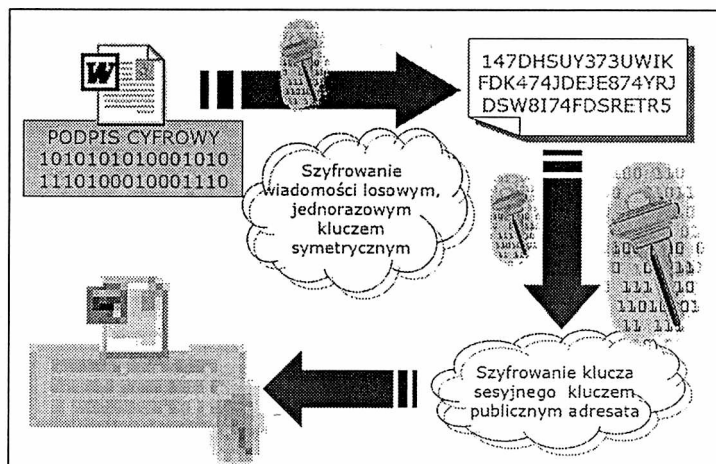
Rys. 4. Składanie podpisu elektronicznego



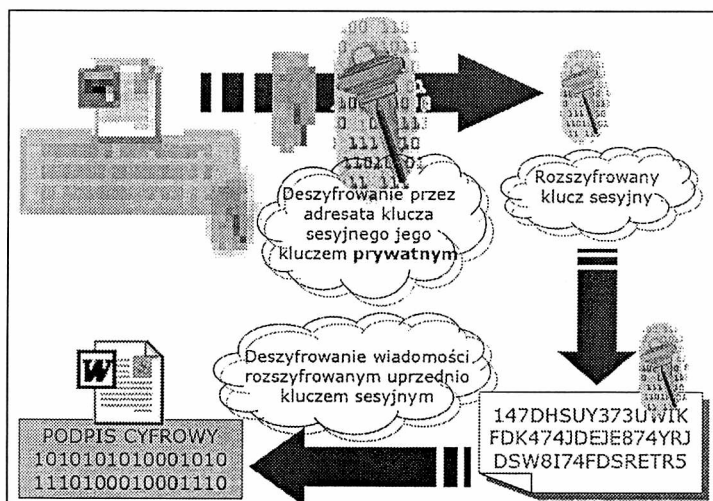
Rys. 5. Weryfikacja podpisu elektronicznego

3.3. SZYFROWANIE I DESZYFROWANIE

Do szyfrowania danych wykorzystuje się specjalnie spreparowany klucz sesyjny (jednorazowy klucz symetryczny generowany dla tego jednego zdarzenia). Następnie klucz ten, szyfrowany jest kluczem publicznym osoby upoważnionej do przeczytania tak zabezpieczonej informacji. Wynikiem operacji szyfrowania jest ciąg bitów reprezentujący oryginalny dokument, do którego dołączany jest zaszyfrowany klucz sesyjny. Algorytm szyfrowania danych przedstawiono na rysunku 6.



Rys. 6. Algorytm szyfrowania danych



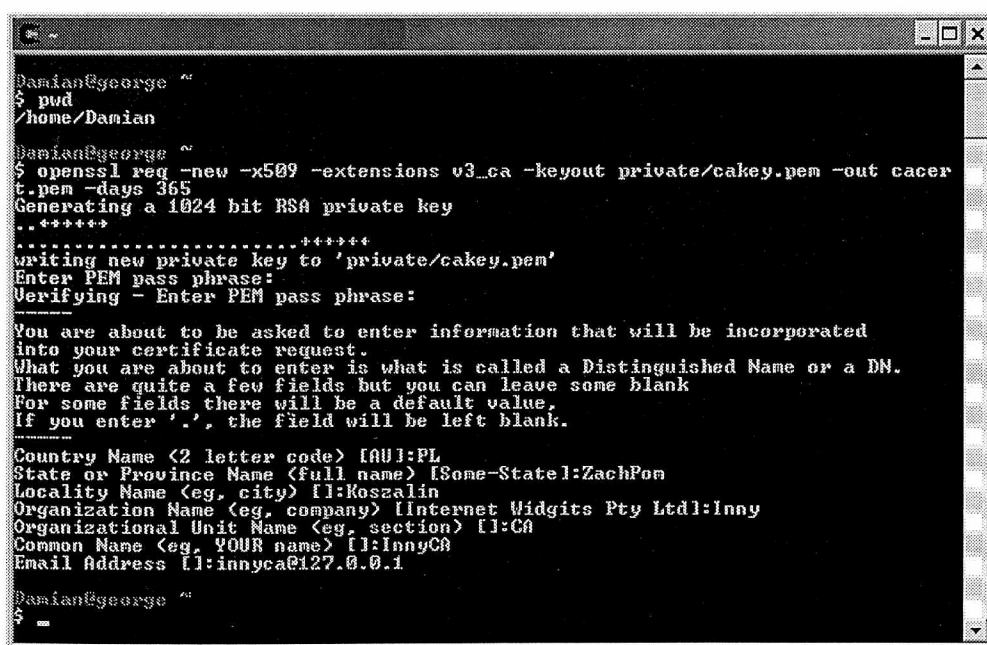
Rys. 7. Algorytm deszyfrowania danych

W przypadku deszyfrowania danych konieczne jest użycie klucza prywatnego, który umożliwi odszyfrowanie uprzednio spreparowanego klucza sesyjnego. Za jego pomocą deszyfrowany jest dokument. Po wykonaniu takiej operacji otrzymujemy dokument pierwotny. Proces deszyfrowania danych przedstawiono na rysunku 7.

4. GENEROWANIE KLUCZY ORAZ CERTYFIKATÓW

W strukturze PKI najważniejszy jest główny Urząd Certyfikacji. Dlatego w pierwszym kroku przygotowujemy dla niego odpowiedni certyfikat i nazwiemy go „InnyCa”. W tym celu wykorzystamy ogólnodostępne i darmowe narzędzie openssl. Warto wspomnieć, iż większość dystrybucji Linuksa posiada go w swoich zasobach. Na rysunku 8 przedstawiono rezultat działania polecenia:

```
„openssl req -new -x509 -extensions v3_ca -keyout private/cakey.pem -out ca-
cert.pem -days 365”
```



```
Damian@george ~
$ pwd
/home/Damian

Damian@george ~
$ openssl req -new -x509 -extensions v3_ca -keyout private/cakey.pem -out ca-
cert.pem -days 365
Generating a 1024 bit RSA private key
.....+++++
.....+++++
writing new private key to 'private/cakey.pem'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name <2 letter code> [AU]:PL
State or Province Name <full name> [Some-State]:ZachPom
Locality Name <eg, city> []:Koszalin
Organization Name <eg, company> [Internet Widgits Pty Ltd]:Inny
Organizational Unit Name <eg, section> []:CA
Common Name <eg, YOUR name> []:InnyCa
Email Address []:innyca@127.0.0.1

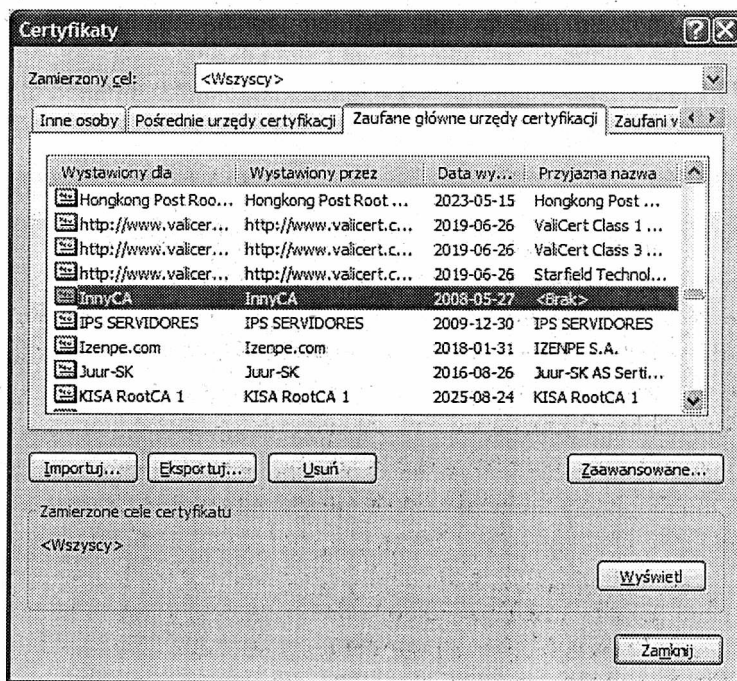
Damian@george ~
$
```

Rys. 8. Generowanie *Root CA*

Rozszerzenie `v3_ca` – oznacza, że przygotowujemy certyfikat *Root CA*, a więc głównego urzędu, którego zadaniem będzie podpisywanie certyfikatów subskrybentów. Pominięto w ten sposób jednostki podległe. Skróci to pokaz, gdzie dodatkowo

musielibyśmy stworzyć certyfikaty kolejnych CA. Będzie to też zgodne z sytuacją w Polsce, bowiem nasze główne CA od razu podpisują certyfikaty klienta końcowego. Jak widać, nie jest to specjalnie rozbudowana struktura. Nasz certyfikat będzie ważny przez rok. Dodatkowo podczas procesu tworzenia certyfikatu została wygenerowana para kluczy. Klucz publiczny zostanie dołączony do certyfikatu, a klucz prywatny zostanie zapisany w osobnym pliku. Klucz ten dodatkowo zabezpieczany jest hasłem PIN (ang. *Personal Identification Number*).

Tak przygotowany certyfikat można zainstalować w repozytorium certyfikatów np. w przeglądarce internetowej, co przedstawiono na rysunku 9. Zauważmy, że jest to jedyny przypadek, gdzie dana jednostka sama sobie wystawia certyfikat (kolumny „Wystawiony dla” i „Wystawiony przez”).



Rys. 9. Repozytorium certyfikatów z dodanym „InnyCA”

W ten sposób staliśmy się właścicielami czwartego w Polsce Centrum Certyfikacji. Po niedługim okresie zgłasza się pierwszy klient. Dobrze, jeśli nasz petent mieszka blisko CA, ale co w przypadku, gdy jest oddalony o wiele kilometrów? Może być jeszcze gorzej, CA wcale nie musi organizować biura, czyli miejsca spotkań z klientem, aby sprawnie wypełniać swoje obowiązki. Może to być część zupełnie innej organizacji. Tak w istocie jest w przypadku trzech polskich CA. Sigillum PCCE to komórka PWPW S.A., Unicert należy do Unizeto S.A., a KIR do KIR S.A. Do zadań

związanych z obsługą klienta wyznaczony jest Urząd Rejestracji (RA). Każde Centrum Certyfikacji może mieć wiele RA. Są to niewielkie biura, gdzie specjalnie przeszkolone osoby zajmują się kompleksową obsługą przyszłego subskrybenta. W każdym Punkcie Rejestracji np. Sigillum PCCE do wystawiania certyfikatów służą specjalnie przygotowane komputery. Chronione są wyjątkowo restrykcyjnymi procedurami. Za ich pomocą generowany jest PKCS#10 (wniosek o certyfikat), a następnie przesyłany do Głównego CA. Ten generuje certyfikat dla wnioskodawcy i odsyła go z powrotem do RA. W kolejnym kroku certyfikat jest zapisywany na karcie kryptograficznej personalizowanej od razu przy kliencie. Cały proces odbywa się w ciągu jednej sesji. Ciekawostką jest, że cała procedura jest podpisywana przez pracownika RA, który również musi posiadać własny podpis elektroniczny. Należy pamiętać, że nie może wystąpić przypadek, w którym osoba do tego nieuprawniona będzie wstanie przeprowadzić taką procedurę. Z drugiej strony, nie może być wątpliwości co do procesu uzyskania takiego certyfikatu, ponieważ straciłby on swoją prawną skuteczność.

```

Damian@George ~
$ openssl req -new -nodes -out inny-req.pem -keyout private/inny-key.pem -days
365
Generating a 1024 bit RSA private key
.....+++++
.....+++++
writing new private key to 'private/inny-key.pem'

You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.

Country Name (2 letter code) [AU]:PL
State or Province Name (full name) [Some-State]:ZachPom
Locality Name (eg. city) []:Koszalin
Organization Name (eg. company) [Internet Widgits Pty Ltd]:Inny
Organizational Unit Name (eg. section) []:Sub
Common Name (eg. YOUR name) []:Inny
Email Address []:inny@127.0.0.1

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:1111
An optional company name []:Inny

Damian@George ~
$

```

Rys. 10. Generowanie PKCS#10

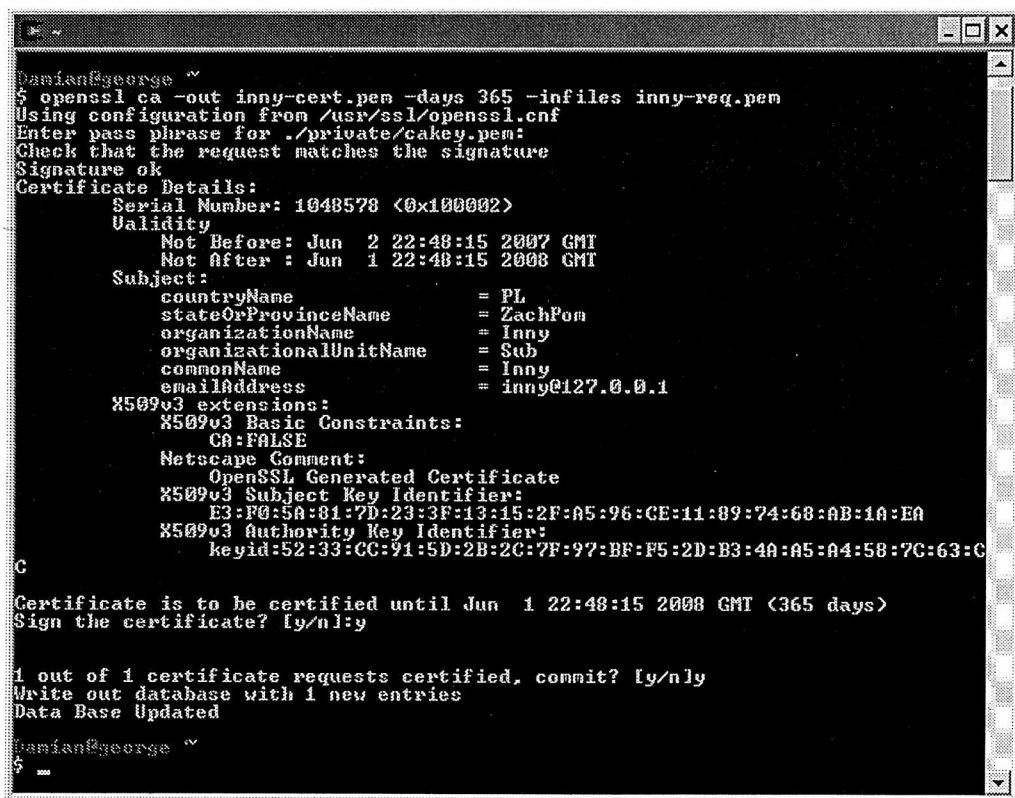
Kolejnym krokiem będzie wygenerowanie wniosku o certyfikat (PKCS#10), który posłuży do uzyskania właściwego certyfikatu. Podobnie wykorzystujemy do tego celu narzędzie openssl i wydajemy polecenie, którego działanie zilustrowano na rysunku 10:

„openssl req -new -nodes -out inny-req.pem -keyout private/inny-key.pem -days 365”

W trakcie tego procesu powstaje para kluczy subskrybenta. Nasz klient nazywa się Inny i posiada adres mail: inny@127.0.0.1. W wyniku pracy programu otrzymujemy dwa pliki. W pierwszym (*out*) program zapisał wniosek o certyfikację, natomiast w drugim (*keyout*) klucz prywatny. Wniosek wysyłamy do CA, który na jego podstawie generuje właściwy certyfikat subskrybenta. Teraz działając w imieniu InnyCA przy użyciu openssl wystawiamy certyfikat poleceniem:

```
„openssl ca -out inny-cert.pem -days 365 -infiles inny-req.pem”
```

Warto zauważyć, że w poleceniu tym nie ma żadnej informacji o lokalizacji repozytorium kluczy głównego CA. Takie informacje zawiera plik konfiguracyjny o rozszerzeniu CNF, o który upomni się aplikacja w trakcie swojej pracy. Takie zachowanie zilustrowano na rysunku 11.



```

Damian@george ~
$ openssl ca -out inny-cert.pem -days 365 -infiles inny-req.pem
Using configuration from /usr/ssl/openssl.cnf
Enter pass phrase for ./private/cakey.pem:
Check that the request matches the signature
Signature ok
Certificate Details:
  Serial Number: 1048578 <0x100002>
  Validity
    Not Before: Jun  2 22:48:15 2007 GMT
    Not After : Jun  1 22:48:15 2008 GMT
  Subject:
    countryName           = PL
    stateOrProvinceName   = ZachPom
    organizationName      = Inny
    organizationalUnitName = Sub
    commonName            = Inny
    emailAddress          = inny@127.0.0.1
  X509v3 extensions:
    X509v3 Basic Constraints:
      CA:FALSE
    Netscape Comment:
      OpenSSL Generated Certificate
    X509v3 Subject Key Identifier:
      E3:F0:5A:81:7D:23:3F:13:15:2F:A5:96:CE:11:89:74:68:AB:1A:EA
    X509v3 Authority Key Identifier:
      keyid:52:33:CC:91:5D:2B:2C:7F:97:BF:F5:2D:B3:4A:A5:A4:58:7C:63:C
C
Certificate is to be certified until Jun  1 22:48:15 2008 GMT (365 days)
Sign the certificate? [y/n]:y

1 out of 1 certificate requests certified, commit? [y/n]:y
Write out database with 1 new entries
Data Base Updated

Damian@george ~
$

```

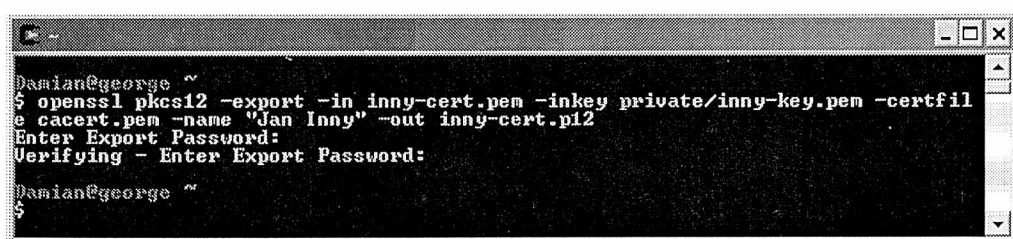
Rys. 11. Generowanie certyfikatu subskrybenta

Otrzymany certyfikat oraz klucz prywatny należy zapisać na kartę kryptograficzną. W zależności od potrzeby subskrybent będzie używał albo klucza prywatnego do podpisywania dokumentów, albo certyfikatu do weryfikacji swojego podpisu lub

po prostu do przedstawiania się w różnego rodzaju systemach informatycznych. Tę sytuację można przyrównać do posiadania ręki oraz dowodu osobistego. Pomimo, że cała procedura została wykonana w kilku krokach, a nie jednej sesji, to została zrealizowana na jednym komputerze, a więc bezpiecznym środowisku. Przynajmniej takim powinno owe środowisko być.

Możliwość importu takiego certyfikatu, na przykład do repozytorium przeglądarki internetowej, zapewni połączenie certyfikatu z kluczem prywatnym. Nie oznacza to jednak połączenia klucza prywatnego z certyfikatem w dosłownym sensie, ponieważ taka zależność występuje tylko w przypadku klucza publicznego. Tutaj raczej chodzi o sprowadzenie całości do wspólnej postaci. Dlatego w następnym kroku zamknijemy całość w formacie PKCS#12, co przedstawiono na rysunku 12.

```
„openssl pkcs12 -export -in inny-cert.pem -inkey private/inny-key.pem -certfile ca-  
cert.pem -name "Jan Inny" -out inny-cert.p12”
```



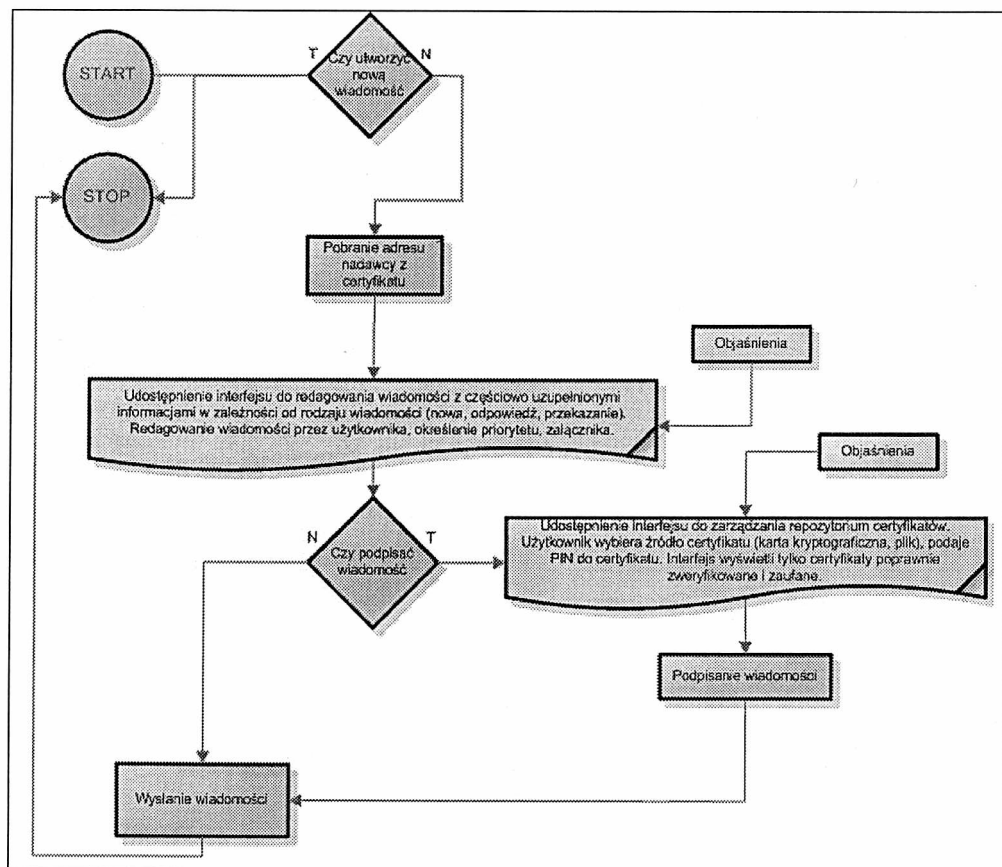
Rys. 12. Eksport certyfikatu oraz klucza prywatnego do PKCS#12

Tak utworzony certyfikat jest postacią końcową. I właśnie taką otrzyma subskrybent od CA. Należy pamiętać, że jest to najprostsza ścieżka generowania certyfikatów. Pomija ona wiele rozszerzeń. Przykładowo certyfikat użytkownika, nie ma utworzonego celu do szyfrowania danych. Tradycyjny klient poczty elektronicznej pomimo, że poprawnie zweryfikuje i doda go do swojego repozytorium, to nie pozwoli zaszyfrować przy jego pomocy wiadomości. Nie dlatego, że jest to niemożliwe – przeciwnie, jak najbardziej jest – ale dlatego, że nie został stworzony do takich celów. Nam jednak wystarczy w zupełności.

5. PODPISYWANIE WIADOMOŚCI ELEKTRONICZNEJ

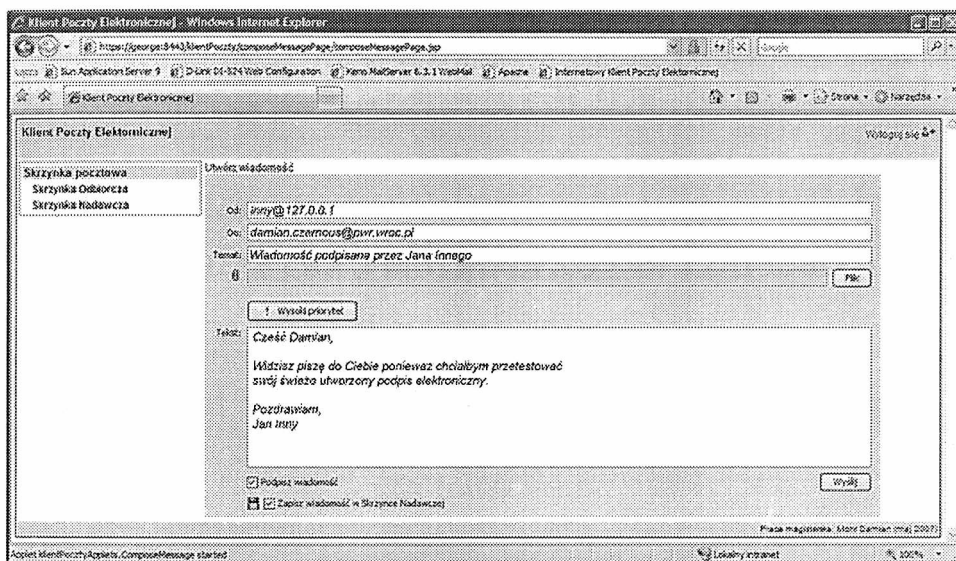
Aby zaprezentować mechanizm podpisywania wiadomości elektronicznej, konieczne jest przygotowanie odpowiedniego środowiska testowego. W tym celu przygotowano aplikację webową osadzoną na serwerze aplikacyjnym firmy SUN. Dodatkowo, potrzebny jest serwer poczty elektronicznej. Aplikacja będzie komuni-

kowała się z nim za pomocą protokołów IMAP oraz SMTP. Sam proces redagowania wiadomości podlega strategii, którą przedstawiono na rysunku 13. Natomiast na rysunku 14 zaprezentowano formularz nadawania nowej wiadomości. W naszym przykładzie Jan Inny (inny@127.0.0.1), będzie korespondował z Damianem Czernousem (damian.czernous@pwr.wroc.pl). Na rysunku 15 zilustrowano działanie appletu podpisującego zredagowaną wiadomość. Wykorzystanie tego typu elementu jest niezbędne w procesie podpisywania. Jest to uwarunkowane tym, że klucz prywatny nie może być wyciągnięty poza swoje repozytorium (polityka bezpieczeństwa), a tym samym nie może być przekazany do serwera, gdzie mógłby być dopiero wykorzystany. Przekazanie klucza poza kartę kryptograficzną lub komputer jego właściciela jest równoznaczne z jego kompromitacją. Applet natomiast jest aplikacją uruchamianą na komputerze lokalnym (nie serwerze), a tym samym może mieć dostęp do jego zasobów i urządzeń. Oczywiście taki applet, również powinien przedstawić się

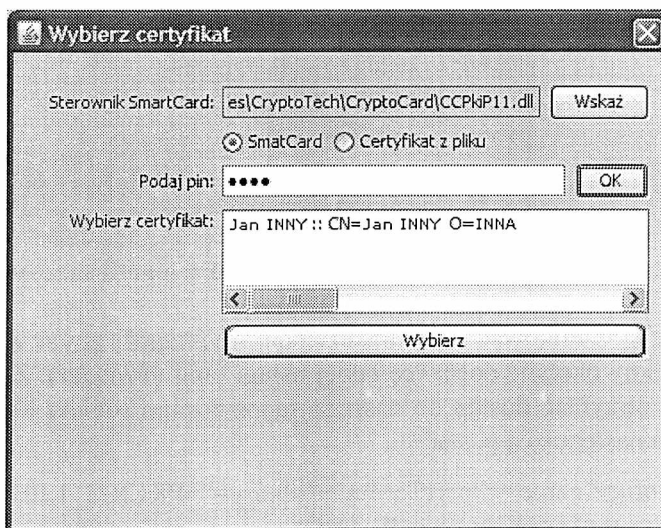


Rys. 13. Algorytm pracy z nową wiadomością

ważnym certyfikatem, któremu będziemy mogli ufać. Aplikacja ta wyśle do karty skrót dokumentu, który zostanie podpisany, a następnie odeśle rezultat do serwera, który dołączy otrzymane dane do naszej wiadomości w postaci załącznika (format PKCS#7). Tak przygotowaną wiadomość będzie można wysłać do adresata.



Rys. 14. Redagowanie nowej wiadomości

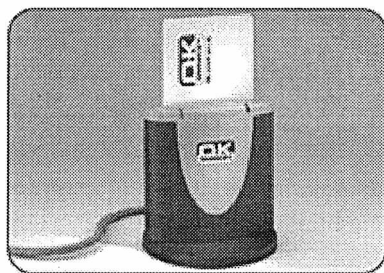


Rys. 15. Apłekt podpisujący – czytanie repozytorium karty kryptograficznej lub pliku

W procesie weryfikacji poprawności podpisanej wiadomości, aplikacja będzie musiała skontaktować się z jego wystawcą w celu odszukania listy CRL lub skorzystać z usługi OCSP, aby móc w pełni zweryfikować ważność podpisu elektronicznego. W przypadku, gdy użytkownik nie wyraził zgody na zamieszczenie swojego certyfikatu w ogólnodostępnym repozytorium certyfikatów, taka weryfikacja jest znacznie utrudniona lub wręcz niemożliwa. Dzieje się tak, ponieważ odbiorca nie ma dostępu do certyfikatu nadawcy. Innymi słowy, zaszyfrowany kluczem prywatnym skrót wiadomości (podpis), może być odszyfrowany kluczem publicznym, który umieszczony jest w certyfikacie, a do którego nie ma dostępu adresat wiadomości. W takim przypadku, jedynym rozwiązaniem jest załączenie takiego certyfikatu do wysyłanej wiadomości przez nadawcę.

5.1. OBSŁUGA REPOZYTORIUM CERTYFIKATÓW

Na rynku dostępnych jest wiele bibliotek do obsługi urządzeń kryptograficznych takich jak czytniki kart kryptograficznych, czy bardziej zaawansowanych HSM (ang. *Hardware Security Module*) np. komercyjne biblioteki IAIK lub darmowe BOUNCYCASTLE. W naszym przypadku skorzystamy z produktu darmowego. W pierwszej kolejności spróbujemy obsłużyć repozytorium znajdujące się na karcie kryptograficznej.



Rys. 16. Czytnik kart kryptograficznych

Każdy producent takiego urządzenia powinien wraz z nim dostarczyć sterownik do jego obsługi.

Aby sprawnie komunikować się z urządzeniem kryptograficznym w pierwszej kolejności powinniśmy określić dostawcę takiej usługi (ang. *provider*). Za jego pomocą będziemy mogli otrzymać dostęp do naszego repozytorium (obiekt klasy *KeyStore*). Zapis w Javie ma następującą postać:

```
String pkcs11Config="name =SmartCard\n"+"library="+PKCS11_LIBRARY_PATH;
byte[] pkcs11Bytes = pkcs11Config.getBytes();
ByteArrayInputStream pkcs11Stream = new ByteArrayInputStream(pkcs11Bytes);
```

```

Class pkcs11Class = Class.forName("sun.security.pkcs11.SunPKCS11");
Constructor pkcs11Constr = pkcs11Class.getConstructor(java.io.InputStream.class);
Provider pkcs11Provider = (Provider) pkcs11Constr.newInstance(pkcs11Stream);
Security.addProvider(pkcs11Provider);
char[] pin = PKCS11_PIN.toCharArray();
KeyStore keystore = KeyStore.getInstance("PKCS11");
keystore.load(null, pin);

```

Z repozytorium możemy pobrać dostępne certyfikaty. Do tego służy metoda `getCertificateChain(String alias)` przyjmująca jako parametr przyjazną nazwę certyfikatu. W rezultacie otrzymamy tablicę obiektów *Certificate*. Jak widać pod jedną nazwą w repozytorium może wystąpić wiele certyfikatów. W naszym jest tylko jeden.

Możemy również otrzymać dostęp do klucza prywatnego. Odpowiedni kod w Javie ma postać:

```

ArrayList<String> aliases = new ArrayList<String>();
ArrayList<Certificate> certificates = new ArrayList<Certificate>();
ArrayList<PrivateKey> privateKey = new ArrayList<PrivateKey>();
Enumeration e = keystore.aliases();
if (e.hasMoreElements()) {
    String tmp = (String) e.nextElement();
    aliases.add(tmp);
    certificates.add(keystore.getCertificateChain(tmp)[0]);
    privateKey.add((PrivateKey) keystore.getKey(tmp, null));
}

```

W przypadku, gdy mamy do czynienia z repozytorium w postaci pliku, sprawa wygląda nieco inaczej. Przede wszystkim odmiennie definiujemy naszego dostawcę usługi, który umożliwia dostęp do klucza prywatnego. Warto zwrócić uwagę, że repozytorium karty kryptograficznej zawiera obiekty formatu PKCS#11, natomiast pliku PKCS#12. Przykładowy kod w Javie wygląda następująco:

```

EncryptionUtils
eU=EncryptionManager.getEncryptionUtils(EncryptionManager.SMIME);
EncryptionKeyManager keyMgr = eU.createKeyManager();

char[] pass = new String(„1234”).toCharArray();
keyMgr.loadPrivateKeystore(new FileInputStream(KEYSTORE_PATH), pass);

Collection col = keyMgr.privateKeyAliases();
Iterator itr = col.iterator();

ArrayList<String> privKey = new ArrayList<String>();
while (itr.hasNext())
    privKey.add(itr.next().toString());

```



```

KeyStore keystore = KeyStore.getInstance("PKCS12");
keystore.load(new FileInputStream(KEYSTORE_PATH), pass);

ArrayList<String> aliases = new ArrayList<String>();
ArrayList<Certificate> certificates = new ArrayList<Certificate>();
Enumeration e = keystore.aliases();
if (e.hasMoreElements()) {
    String tmp = (String) e.nextElement();
    aliases.add(tmp);
    certificates.add(keystore.getCertificateChain(tmp)[0]);
}

```

6. PODSUMOWANIE

Technologie związane z wykorzystaniem infrastruktury PKI nie są nowością. Przeciwnie, istnieją od wielu lat i są coraz bardziej udoskonalane zachowując swój wysoki poziom bezpieczeństwa. Jednak ta gałąź informatyki wiąże się ściśle z przepisami prawnymi. Jeśli nie będą sprawnie funkcjonować właściwe przepisy prawne, cała technologia staje się niejako fantastyką, bowiem nie można z niej korzystać.

Powoli rosnąca popularność certyfikatów stanie się tematem wielu kontrowersyjnych dyskusji, podobnie jak miało to miejsce, gdy wdrażano karty bankomatowe. Jednak społeczeństwo przyzwyczai się do ich istnienia, aż staną się codziennością. Jest to na razie przyszłość, choć pierwsze kroki zostały już poczynione. Jednostki administracji publicznej, od 1 maja 2008 r. (DzU 2006 nr 145 poz. 1050), powinny obsługiwać napływające sprawy drogą elektroniczną, a więc za pośrednictwem Elektronicznej Skrzynki Podawczej. Interesant musi jednak być wyposażony w kwalifikowany podpis elektroniczny, który nie jest wcale tanim zakupem. Rozwiązania teleinformatyczne są często wadliwe, a przez to są przedmiotem wielu krytyk znawców tematu. Główną ich wadą jest brak umiejętności wzajemnej współpracy, co na większą skalę oznacza ich bezużyteczność. Wina za ten stan rzeczy leży po stronie ustawodawcy, który nie określił jednoznacznie potrzebnych standardów (formatu podpisu, formatu dokumentu, sposobu komunikacji) pozostawiając tym samym dużą swobodę rywalizującym ze sobą firmom. Nie dopracował również przepisów, które często przeczą sobie nawzajem. Wciąż nie uchwalono nowelizacji Ustawy o podpisie elektronicznym. Usunięto jedynie kilka luk prawnych, które poprawiając kwestie standardów, dostarczały nowych wątpliwości. Wreszcie, nie jest to temat specjalnie nagłaśniany. Wiele osób nie wie, gdzie mogłyby taki certyfikat nabyć, albo gdzie uzyskać odpowiednie informacje. Sprawnym rozwiązaniem w zakresie powszechności podpisu elektronicznego jest jego dostarczenie wraz z dowodem osobistym każdemu obywatelowi, który osiąga pełnoletniość. Z pewnością w takiej sytuacji doszłoby do upo-

wszechnienia certyfikacji elektronicznej. Dopóki jednak nie ma odbiorcy usługi, nie ma potrzeby jej dostarczania.

Obecnie podpis elektroniczny wykorzystywany jest przez różne podmioty gospodarcze np. w przetargach elektronicznych. I póki co, to właśnie im najbardziej zależy na jego sprawnym funkcjonowaniu. Można wymienić wiele zastosowań, w których ludzie chętnie zawieraliby umowy między sobą lub z instytucjami (np. bankami, pracodawcami), przez komputer, gdzie łatwa autoryzacja do różnorodnych systemów teleinformatycznych byłaby codziennością (koniec z pamiętaniem haseł i loginów). Wykorzystanie potencjału podpisu elektronicznego wiąże się z redukcją kosztów na olbrzymią skalę i jest jak najbardziej ideą proekologiczną (eliminacja papieru – podpis elektroniczny jest sprawny jedynie w swoim cyfrowym środowisku). Biorąc to pod uwagę, aż dziw bierze, że temat ten wciąż jest tak mało popularny.

Na koniec zauważmy, że aspekt technologiczny nie nadaje się na pracę badawczą, ponieważ wykorzystuje on dobrze znane algorytmy oraz mechanizmy. Zatem akademickie prace zostały ukończone, pozostało tylko korzystać ze zgromadzonej wiedzy. To od nas zależy, kiedy zawód grafologa stanie się ciekawostką (tak jak repasaczka pończoch) a podpis elektroniczny, będzie mógł zająć w historii ludzkości zasłużone miejsce obok pierwszej maszyny drukarskiej.

LITERATURA

- [Ustawa 2001] *Ustawa o podpisie elektronicznym*, DzU 01.130.1450, Warszawa 2001.
- [Czernous 2007a] Czernous D., *Internetowy Klient Poczty Elektronicznej z wykorzystaniem mechanizmów PKI*, Koszalin 2007.
- [Czernous 2007b] Czernous D., *Podpis elektroniczny, faktura elektroniczna, aukcja elektroniczna uwa-runkowania prawne i rozwiązania*, Koszalin 2007.

DIGITAL SIGNATURE THE SIGN OF NEW EDGE

This article is focused on digital signature. It's concentrated on technological, law and psychology aspect. Because of that, this is addressed to every recipient of no account on specialization, which is curiosity of this theme. On particular attention deserve chapter "Digital Signature", which explains what is it mean to sign electronic document. Those, who are developers should be interested with handling certificates database placed on smart card or file (code language: Java). Whole has clear form, presents interesting points of view, richly illustrated and full of examples.

*aktywne bazy danych,
reguły ECA,
SZBD Oracle i MS SQL Server*

Zbigniew STASZAK*
Dawid JASIŃSKI

PORÓWNANIE MOŻLIWOŚCI TWORZENIA AKTYWNYCH BAZ DANYCH W SZBD ORACLE I MICROSOFT SQL SERVER

Praca w pierwszej części prezentuje zagadnienia teoretyczne związane z aktywnymi bazami danych. W drugiej części przedstawione są mechanizmy tworzenia takich baz w SZBD Oracle 11g i Microsoft SQL Server 2005. W kolejnej części przeprowadzone jest porównanie tych mechanizmów w obu SZBD. Analiza porównawcza przeprowadzona jest na podstawie testów, którym poddana została przykładowa aktywna baza danych zaimplementowana w obu SZBD. Na zakończenie przedstawiono wnioski.

1. WPROWADZENIE

Aktywna baza danych, w odróżnieniu od bazy pasywnej, posiada wbudowane mechanizmy, które pozwalają jej w sposób inteligentny reagować na zachodzące zdarzenia w bazie i wokół bazy. Zdarzenia te można podzielić na trzy rodzaje [Mazur 2003] (rys. 1). Pierwszym z nich są zdarzenia oparte o czas i interwały czasowe. Przykładem takiego zdarzenia jest tworzenie zapasowej kopii danych o określonej godzinie i w określonym dniu lub tworzenie raportu, co ustaloną liczbę nowych wpisów do bazy danych. Drugim rodzajem są zdarzenia użytkownika, które charakteryzują się tym, że za ich skonstruowanie odpowiada użytkownik. Przykładem takiego zdarzenia jest aktualizacja krotek relacji. Trzecim rodzajem zdarzeń są zdarzenia w bazie danych. Zdarzenia te mogą zostać wygenerowane przy wykonywaniu akcji zdarzenia zaobserwowanego w bazie danych. Dzieje się tak, gdy realizacja akcji powoduje modyfikację bazy danych, która prowadzi do wywołania zdarzenia przechwytywanego przez bazę. Przykładem może być sytuacja, w której użytkownik zgłasza chęć dokonania przelewu pewnej

* Instytut Informatyki, Wydział Informatyki i Zarządzania Politechniki Wrocławskiej, 50-370 Wrocław, Wybrzeże Wyspiańskiego 27, zbigniew.staszak@pwr.wroc.pl

2003, Koszłajda 2007])). Do pierwszej z nich należą zdarzenia związane ze stanem obiektu. Do drugiej kategorii należy zaliczyć zdarzenia związane ze zmianami stanu transakcji. Trzecia kategoria gromadzi zdarzenia temporalne. Do czwartej kategorii należą zdarzenia zdefiniowane przez użytkownika. W ostatniej, piątej kategorii występują zdarzenia obserwowane w systemie zarządzającym bazą danych. Poniżej przedstawiono operacje wykonywane w ramach każdej z kategorii.

- Operacje dla zdarzeń związanych ze stanem obiektu
 - a) tworzenie obiektu (AFTER CREATE)
 - b) usuwanie obiektu (BEFORE DROP)
 - c) modyfikacja obiektu ({BEFORE|AFTER} ALTER)
 - d) odczyt obiektu ({BEFORE|AFTER} READ)
 - e) dostęp do obiektu ({BEFORE|AFTER} ACCESS)
- Operacje związane ze zmianami stanu transakcji
 - a) rozpoczęcie transakcji (AFTER TBEGIN)
 - b) określenie punktu zatwierdzenia transakcji (AFTER TCOMPLETE)
 - c) zatwierdzenie transakcji (AFTER TCOMMIT)
 - d) wycofanie transakcji ({BEFORE|AFTER} TROLLBACK)
- Operacje związane ze zdarzeniami temporalnymi
 - a) określające punkt czasu (AT TIME (timestamp))
 - b) występujące po upływie czasu (AFTER TIME (interval))
 - c) okresowe, co określony czas (EVERY TIME (interval))
- Operacje związane ze zdarzeniami zdefiniowanymi przez użytkownika
 - a) zdefiniowane przez użytkownika
({BEFORE|AFTER} nazwa_użytkownika)
- Operacje dotyczące SZBD
 - a) autoryzacja do SZBD (LOGIN, LOGOUT)
 - b) uruchomienie bazy danych (STARTUP, SHUTDOWN)
 - c) błąd (ERROR)

Obsługa zdarzeń elementarnych odbywa się w chwili ich wystąpienia. Zdarzeniu można przyporządkować interwał czasowy. Konkretny interwał obliczany jest jako różnica czasu wystąpienia zdarzenia i czasu zakończenia zdarzenia. W zdarzeniach prostych wartość czasu wystąpienia jest równa wartości czasu zakończenia. Interwał czasowy zdarzenia elementarnego wynosi więc zero. W opisywanych, w dalszej części, zdarzeniach złożonych, interwał czasowy jest często większy zera. Interwał czasowy wykorzystywany jest do identyfikacji zdarzeń, które miały prawo pojawić się w systemie.

3. ZDARZENIA ZŁOŻONE

W rzeczywistym świecie istnieją sytuacje, których nie można modelować tylko za pomocą zdarzeń prostych. Konstruowane są wtedy, wykorzystujące zdarzenia proste i operatory zdarzeniowe, zdarzenia złożone. Zdarzenie te mogą być także budowane w oparciu o inne, wcześniej zdefiniowane, zdarzenia złożone. Zdarzenie złożone składa się ze zdarzeń pośrednich i można je na nie podzielić [Chwalisz 2005]. Wyróżnia się trzy etapy wystąpienia zdarzenia złożonego: inicjacja, zaobserwowanie, zakończenie. Na interwał czasowy zdarzenia złożonego składa się suma interwałów zdarzeń pośrednich. Czas rozpoczęcia t_R zdarzenia złożonego określony jest przez czas rozpoczęcia t_R zdarzenia inicjującego. Z kolei czas zakończenia t_Z zdarzenia złożonego przyjmując taką samą wartość jak czas zakończenia t_Z zdarzenia kończącego.

4. WYRAŻENIA ZDARZENIOWE

Zdarzenie identyfikowane jest za pomocą czasu jego zaobserwowania w systemie. Każde zdarzenie posiada przypisany unikatowy znacznik czasowy, który służy do umieszczenia go wśród innych zdarzeń, zgodnie z czasem jego wystąpienia. Z regułą aktywną powiązana jest indywidualna historia zdarzeń, która pomocna jest przy określaniu kolejności wystąpienia zdarzeń. Wyrażenia zdarzeniowe pozwalają, przy wykorzystaniu operatorów zdarzeniowych, definiować zdarzenia proste oraz złożone. Zdarzenia proste są zdarzeniami dostarczonymi przez SZBD i mają charakter predefiniowany. Istnieje możliwość przeddefiniowania akcji przypisanych do aktywnych reguł opisujących takie zdarzenie. Zdarzenia złożone są zdarzeniami skonstruowanymi przez użytkownika [Mazur 2003]. Dziedzina i przeciwdziedzina wyrażenia zdarzeniowego (E) jest historia zdarzeń (h). Oznacza to, że $E: h \rightarrow h$. W celu określenia wartości zdarzenia złożonego, należy zweryfikować wartości poszczególnych zdarzeń składowych. Analiza zdarzeń złożonych odbywa się wyłączenie na podstawie historii zdarzeń, więc historia ta musi być pamiętana [Koszlańda 2007].

5. OPERATORY ZDARZENIOWE

Operatory zdarzeniowe służą do konstruowania zdarzeń złożonych ze zdarzeń prostych i/lub złożonych. Tworzenie takich zdarzeń odbywa się w sposób rekurencyjny. Wartość interwału czasowego zdarzenia złożonego uwarunkowana jest operatorami użytymi w definicji aktywnej reguły. Każdy z operatorów posiada odmienną semantykę decydującą o różnej wartości interwałów czasowych. Operatory przedstawione poniżej nie określają całego ich zbioru. Nie opisano operatorów rekurencyjnych, poto-

kowych oraz niektórych odmian operatorów okien. W zastosowaniach komercyjnych, zazwyczaj, zbiór operatorów ograniczony jest do operatora logicznego OR.

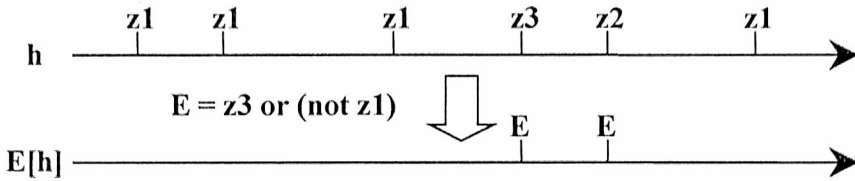
Operatory zdarzeniowe dzieli się na ([Chwalisz 2005, Koszłajda 2007, Adaikkalavan 2002]):

- operatory logiczne
 - AND
 - OR
 - NOT
- operatory następstwa
 - sekwencji (sequence)
 - prior
 - relative
 - PLUS
- operatory wyliczeniowe
 - niecykliczny
 - cykliczny every
- operator okna (firstAfter)
- operator aperiodyczny
- operator periodyczny

Operator logiczny AND określa zależności czasowe pomiędzy pojawieniem się zdarzeń prostych E1 i E2, z których tworzone jest zdarzenie złożone. Zdarzenia E1 i E2 muszą zostać zaobserwowane w określonym czasie.

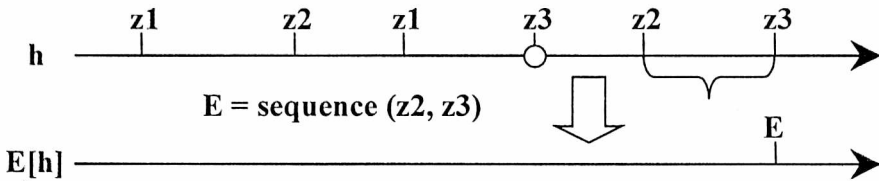
Dla operatora logicznego OR i zdarzeń prostych E1 i E2 warunek wystąpienia zdarzenia złożonego będzie spełniony, gdy zaobserwowane zostanie przynajmniej jedno ze zdarzeń składowych.

Operator logiczny NOT dla zdarzenia prostego E1 oznacza, że zdarzenie to nie może wystąpić. Jeżeli warunek zdarzenia złożonego składa się z różnych zdarzeń i operatorów, a w definicji warunku został użyty operator NOT dla zdarzenia E1, wówczas brak zaobserwowania takiego zdarzenia spowoduje spełnienie tej części warunku zdarzenia złożonego. Przykład wyrażenia zdarzeniowego opartego na operatorach logicznych zilustrowano na rysunku 2.

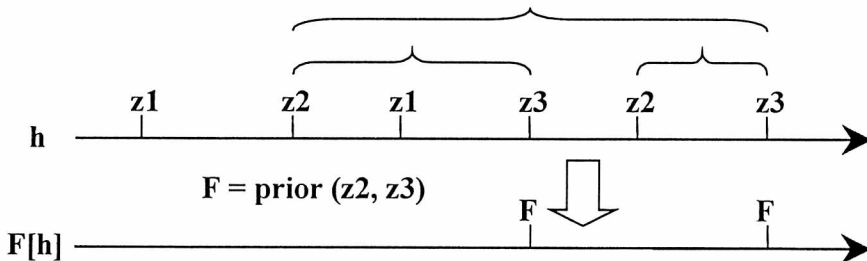


Rys. 2. Przykład wykorzystania operatorów logicznych [Koszłajda 2007]

Operator *sequence* dla zdarzeń elementarnych E_1 i E_2 określa, że zdarzenie E_2 może wystąpić tylko bezpośrednio po zaistnieniu zdarzenia E_1 . Nie mogą więc istnieć żadne inne zdarzenia, które zostałyby zaobserwowane między zdarzeniami E_1 i E_2 . Operator ten określa kolejność następowania zdarzeń po sobie (rys. 3).

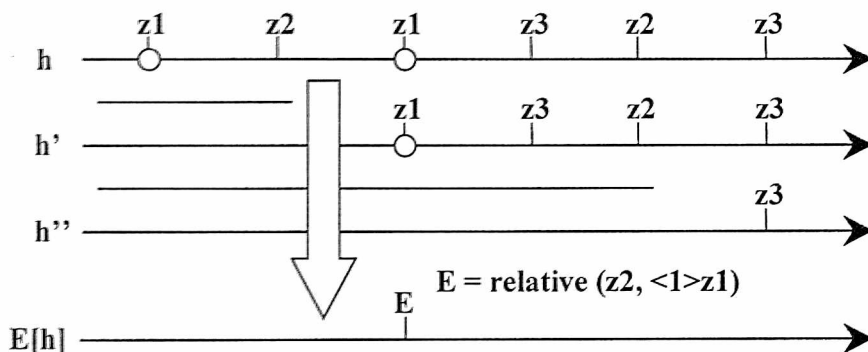
Rys. 3. Przykład operatora *sequence* [Koszłajda 2007]

Operator *następstwa prior* dla zdarzeń prostych E_1 i E_2 oznacza, że pomiędzy wystąpieniem zdarzenia E_1 i E_2 może istnieć dowolna liczba innych zdarzeń. Jeżeli zaistnieje zdarzenie E_1 , to po nim mogą zostać zaobserwowane różne zdarzenia, ale inne niż E_1 i E_2 . Liczba zdarzeń pomiędzy E_1 i E_2 jest dowolna, więc w szczególnym przypadku może również wynieść zero. Wtedy operator *prior* przyjmuje definicję operatora *sequence*. Operator *sequence* jest, więc szczególnym przypadkiem operatora *prior* (rys. 4).

Rys. 4. Przykład operatora *prior* [Koszłajda 2007]

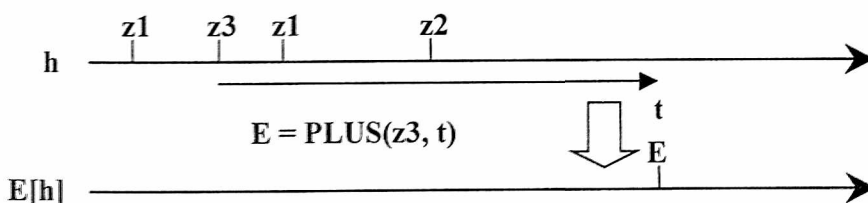
Operator *relative* jest również operatorem *następstwa*, jednak ma odmienną semantykę od operatora *sequence* i operatora *prior*. Operator *relative* wpro-

wadza następstwo dla zdarzeń prostych $E1$ i $E2$. Jeżeli w historii zdarzeń h wystąpi zdarzenie $E2$, to jest tworzona nowa historia zdarzeń h' przez usunięcie z h zdarzeń występujących przed zdarzeniem $E2$ wraz z nim. Zdarzenie złożone składające się ze zdarzeń $E1$, $E2$ i operatora *relative* zostanie zaobserwowane, gdy zdarzenie $E1$ wystąpi w obu historiach zdarzeń: h i h' z tym samym znacznikiem czasowym (rys. 5).



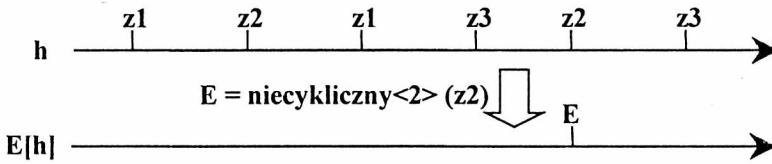
Rys. 5. Przykład operatora *relative* [Koszłajda 2007]

Argumentem operatora *PLUS* [Adaikkalavan 2002] dla zdarzenia prostego $E1$ jest dodatkowo czas t , który służy do ustalenia momentu wywołania zdarzenia złożonego. Jeżeli zostanie zaobserwowane wystąpienie zdarzenia elementarnego $E1$, to po upływie czasu t , nastąpi wywołanie zdarzenia złożonego opartego na operatorze *PLUS* (rys. 6).



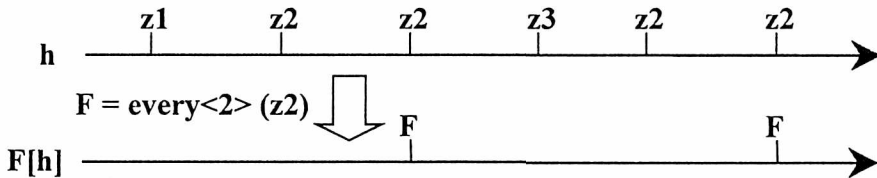
Rys. 6. Przykład operatora *PLUS*

Operator wyliczeniowy niecykliczny jest definiowany dla zdarzenia prostego $E1$ i liczby naturalnej n . Liczba naturalna n oznacza, które wystąpienie zdarzenia elementarnego $E1$ ma spowodować wywołanie zdarzenia złożonego. W całej historii zdarzeń zdarzenie złożone może być wywołane za pomocą operatora wyliczeniowego niecyklicznego co najwyżej raz (rys. 7).

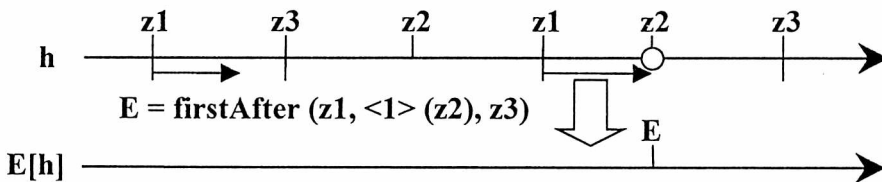


Rys. 7. Przykład operatora niecykliczny [Koszłajda 2007]

Operator wyliczeniowy cykliczny *every* jest również definiowany dla zdarzenia prostego $E1$ i liczby naturalnej n . Tym razem liczba naturalna n dostarcza informację o tym, co które wystąpienie zdarzenia elementarnego $E1$ ma być wywołane zdarzenie złożone (rys. 8).

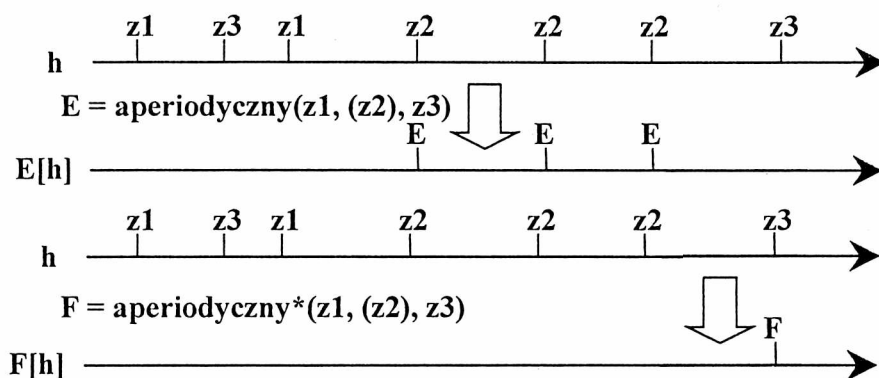
Rys. 8. Przykład operatora cyklicznego *every* [Koszłajda 2007]

Operator okna *firstAfter* oparty jest o trzy zdarzenia proste: $E1$, $E2$ i $E3$. Zdarzenia elementarne $E1$ i $E3$ stanowią rodzaj okna logicznego. Oznacza to, że oczekiwane zaobserwowanie zdarzenia $E2$ ma nastąpić między wystąpieniami zdarzeń $E1$ i $E3$ (rys. 9).

Rys. 9. Przykład operatora *firstAfter* [Koszłajda 2007]

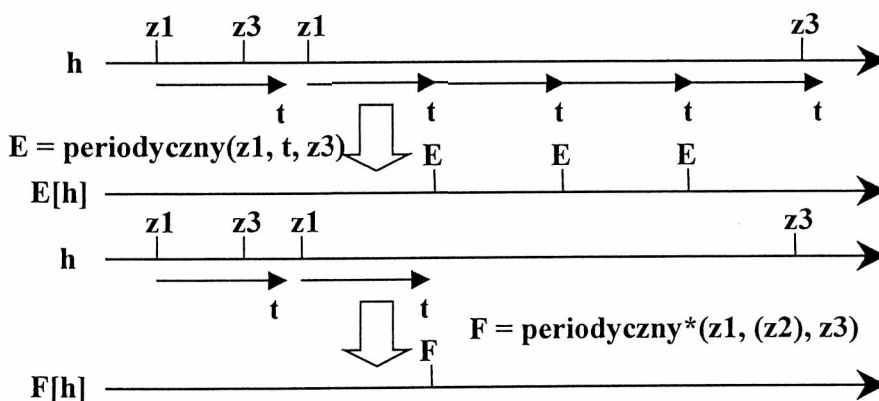
Operator aperiodyczny [Adaikkalavan 2002] opisywany jest za pomocą trzech zdarzeń prostych: $E1$, $E2$ i $E3$. Istnieją dwa jego rodzaje: z kumulacją i bez kumulacji [Chwalisz 2005]. Zdarzenie $E1$ jest zdarzeniem inicjującym operator, natomiast $E3$ kończącym. Dla operatora bez kumulacji pojawienie się zdarzenia elementarnego $E2$ powoduje za każdym razem wywołanie zdarzenia złożonego. W przypadku operatora z kumulacją, wszystkie zdarzenia $E2$ traktowane są jako jedno zdarzenie i wywołanie zdarzenia złożonego nastąpi tylko raz w momencie zaobserwowania zdarzenia kończącego $E3$. Operator aperiodyczny, podobnie

jak `firstAfter`, wprowadza okno logiczne, wewnątrz którego może zaistnieć zdarzenie elementarne E_2 (rys. 10).



Rys. 10. Przykład operatora aperiodyczny z kumulacją oraz bez kumulacji

Operator `periodyczny` [Adaikkalavan 2002] opisany jest za pomocą dwóch zdarzeń elementarnych: E_1 i E_3 oraz czasu t , który musi upłynąć pomiędzy wystąpieniem zdarzenia E_1 i E_3 [Chwalisz 2005]. Zdarzenie E_1 jest zdarzeniem inicjującym operator, natomiast E_3 kończącym. Podobnie do operatora aperiodycznego istnieją jego dwa rodzaje: z kumulacją i bez kumulacji. Operator z kumulacją wywoła tylko raz zdarzenie złożone, jeśli upłynie czas t . Natomiast operator bez kumulacji wywoła zdarzenie złożone za każdym razem, gdy upłynie czas t . Zdarzenia E_1 oraz E_3 stanowią rodzaj okna logicznego, w którym wykonywana jest akcja po określonym czasie (rys. 11).



Rys. 11. Przykład operatora periodyczny z kumulacją oraz bez kumulacji

6. PRZYKŁAD AKTYWNEJ BAZY DANYCH

Aktywne bazy danych bardzo dobrze sprawują się przy modelowaniu procesów przemysłowych i procesów zarządzających urządzeniami. Do testów mechanizmów aktywności zamodelowany został proces technologiczny wytwarzania rozdzielnic średniego oraz niskiego napięcia w firmie Elektromonraż-Export S.A. Na tym przykładzie zaprezentowano możliwe aktywności, jakie baza powinna realizować.

Aktywność bazy danych w głównej mierze zostanie skupiona wokół relacji zawierającej informacje o kolejności wykonywanych zadań oraz informację o hierarchicznej zależności zadań od innych zadań. Relacja ta swoją strukturą przypomina harmonogram zadań, jaki należy wykonać, aby skonstruować rozdzielnicę. Informacje zawarte w relacji określają czy zadanie może być wykonywane w sposób współbieżny, czy też rozpoczęcie wykonywania zadania musi być odłożone do momentu zakończenia realizacji innego zadania (zależnego).

Poniżej przedstawiono opisy eksperymentów wykonanych na przykładowej bazie danych zaimplementowanej zarówno w SZBD Oracle 11g, jak i w SZBD Microsoft SQL Server 2005. Eksperymenty te posłużyły do przetestowania w praktyce możliwości wykorzystania zdarzeń prostych oraz operatorów zdarzeniowych do konstrukcji zdarzeń złożonych.

Eksperymenty wykonywane w sposób manualny (montaż elementów elektrycznych i konstrukcyjnych wykonywany przez pracowników)

EKS001

<i>Krótki opis:</i>	Brak potrzebnych materiałów.
<i>Zdarzenie wyzwalające:</i>	Zgłoszenie pracownika o braku materiałów do realizacji zadania. Przejście zadania ze stanu „wykonywane” do stanu „zatrzymane”.
<i>Przewidywana reakcja:</i>	Przydzielenie pracownikowi innego zadania do wykonania. Poinformowanie kierownika o zatrzymaniu zadania i o braku potrzebnych surowców.

EKS002

<i>Krótki opis:</i>	Aktualizacja danych personalnych.
<i>Zdarzenie wyzwalające:</i>	Zmiana danych personalnych pracowników fizycznych lub kierowników.
<i>Przewidywana reakcja:</i>	Zgodnie z ustawą o ochronie danych osobowych następuje archiwizacja dokonanych zmian wraz z dodatkowymi informacjami: kto dokonał zmian, jakiego rodzaju były to zmiany, kiedy one nastąpiły.

EKS003*Krótki opis:*

Wznowienie wykonywania zadania.

Zdarzenie

Odnutowanie dostarczenia materiałów do magazynu.

*wyzwalające:**Przewidywana*

Określenie, które z zatrzymanych zadań potrzebują tego rodzaju materiału. Następnie wznowienie wykonywania zatrzymanych zadań.

*reakcja:***EKS004***Krótki opis:*

Wykonanie kolejnego zadania.

Zdarzenie

Pracownik informuje o pomyślnie zakończonej realizacji zadania. Stan zadania został określony jako „gotowe”. Czas przeznaczony na wykonanie danego zadania nie upłynął.

*wyzwalające:**Przewidywana*

Określenie, które zadanie ze zbioru zadań może być wykonane niezależnie od innych i przydzielenie tego zadania pracownikowi.

*reakcja:***EKS005***Krótki opis:*

Opóźnienie w realizacji zadania.

Zdarzenie

Czas realizacji dla określonego zadania upłynął.

*wyzwalające:**Przewidywana*

Zostaje sporządzona odpowiednia informacja, a następnie dostarczona kierownikowi. Odnutowane jest również kolejne opóźnione zadanie przez pracownika i jeżeli poprzednie wykonywane zadanie też było opóźnione, wówczas ta informacja też zostanie uwzględniona.

*reakcja:***EKS006***Krótki opis:*

Opóźnienie w realizacji zamówienia.

Zdarzenie

Czas realizacji dla określonego zamówienia upłynął.

*wyzwalające:**Przewidywana*

Sporządzona zostaje odpowiednia informacja, a następnie dostarczona kierownikowi.

*reakcja:***EKS007***Krótki opis:*

Przydzielenie kolejnego zadania.

Zdarzenie

Otrzymanie zadania do wykonania. Pracownik informuje, że rozpoczął realizowanie otrzymanego zadania. Stan zrealizowanego zadania został określony jako „wykonywane”.

*wyzwalające:**Przewidywana*

Sporządzona zostaje określona informacja na temat rozpoczętego zadania i dostarczona kierownikowi. Zostaje uruchomione zdarzenie czasowe, które wykona się w terminie realizacji, informując o opóźnieniu, gdy zadanie nie będzie wykonane na czas.

reakcja:

EKS008*Krótki opis:*

Zrealizowanie określonego zadania.

*Zdarzenie**wyzwalające:*

Wykonanie zadania przez pracownika. Pracownik informuje, że zakończył realizowanie otrzymanego zadania. Stan zrealizowanego zadania został określony jako „gotowe”.

Przewidywana reakcja:

Stosowna informacja zostaje przekazana kierownikowi.

EKS009*Krótki opis:*

Zakończona realizacja zamówienia.

*Zdarzenie**wyzwalające:*

Czas realizacji dla określonego zamówienia nie upłynął. Dodatkowo wszystkie zadania związane z tym zamówieniem zostały wykonane.

Przewidywana reakcja:

Informacja o pomyślnej realizacji, zawierająca numer zamówienia oraz nazwę rozdzielnicę, zostaje przekazana kierownikowi. Stan zrealizowanego zamówienia określony będzie jako „zrealizowane”.

EKS010*Krótki opis:*

Dostarczenie materiałów do magazynu.

*Zdarzenie**wyzwalające:*

Odnotowany został wzrost składowanych materiałów w magazynie.

Przewidywana reakcja:

Sporządzona informacja, określająca, jaki materiał oraz ile sztuk materiału dotarło do magazynu, zostaje dostarczona kierownikowi.

EKS011*Krótki opis:*

Zamówienie materiałów.

*Zdarzenie**wyzwalające:*

Zaobserwowany został kolejny wpis do relacji przechowującej zamówione materiały.

Przewidywana reakcja:

Informacja zawierająca nazwę materiału, liczbę sztuk oraz dane osoby zamawiającej zostaje przekazana kierownikowi. Zostaje uruchomione zdarzenie czasowe, które wykona się w terminie realizacji dostarczenia materiałów, informując o opóźnieniu, gdy materiały te nie dotrą na czas.

EKS012*Krótki opis:*

Opóźnienie w dostawie materiału.

*Zdarzenie**wyzwalające:*

Czas przeznaczony na dostarczenie odpowiedniej liczby sztuk zamówionego materiału upłynął.

Przewidywana reakcja:

Sporządzona zostaje odpowiednia informacja, a następnie przekazana kierownikowi. Odnotowane jest również kolejne opóźnienie dostawy materiałów przez określoną firmę.

EKS013*Krótki opis:*

Rozpoczęcie realizacji zamówienia.

Zdarzenie

Stan zamówienia został zmieniony ze stanu „oczekujące” na „wykonywane”.

*wyzwalające:**Przewidywana reakcja:*

Informacja o rozpoczęciu realizacji zamówienia jest dostarczona do kierownika. Zostaje uruchomione zdarzenie czasowe, które wykona się w terminie realizacji, informując o opóźnieniu, gdy zamówienie nie będzie zrealizowane na czas.

EKS014*Krótki opis:*

Zatwierdzenie realizacji zamówienia.

Zdarzenie

Zaobserwowanie zatwierdzenia zamówienia.

*wyzwalające:**Przewidywana reakcja:*

Sporządzona zostaje informacja o zatwierdzeniu zamówienia zawierająca dane personalne osoby zatwierdzającej oraz numer zamówienia, a następnie przekazana jest kierownikowi. Stan zamówienia został zmieniony ze stanu „oczekujące” na „wykonywane”.

EKS015*Krótki opis:*

5-krotne opóźnienie w dostawie materiałów.

Zdarzenie

Zaobserwowanie piątego zdarzenia nie dostarczania materiałów w określonym terminie przez konkretną firmę.

*wyzwalające:**Przewidywana reakcja:*

Sporządzona zostaje informacja zawierająca sugestię dotyczącą zmiany dostawcy, a następnie dostarczona jest kierownikowi.

EKS016*Krótki opis:*

3-krotne opóźnienie realizacji zadania.

Zdarzenie

Zaobserwowanie trzeciego z kolei opóźnienia zadania przez pracownika.

*wyzwalające:**Przewidywana reakcja:*

Sporządzona zostaje informacja zawierająca sugestię dotyczącą wzmożonej kontroli pracownika, a następnie dostarczona jest kierownikowi.

EKS017*Krótki opis:*

Zmniejszenie ceny realizacji zamówienia.

Zdarzenie

Realizacja zamówienia opóźniła się.

*wyzwalające:**Przewidywana reakcja:*

Zostaje zmniejszona cena realizacji zamówienia o 3%.

Eksperymenty wykonywane w sposób automatyczny (malowanie elementów konstrukcyjnych oraz początkowe prace konstrukcyjne przy wykorzystaniu specjalistycznych maszyn, gdzie głównym surowcem jest blacha ocynkowana)

EKS018

<i>Krótki opis:</i>	Archiwizacja zamówień.
<i>Zdarzenie</i>	Polecenie wyłączenia bazy danych.
<i>wyzwalające:</i>	
<i>Przewidywana reakcja:</i>	Sporządzana jest odpowiednia informacja o planowanej archiwizacji krotek relacji zamówienia, a następnie dokonywana jest archiwizacja.

EKS019

<i>Krótki opis:</i>	Wykonanie kolejnego zadania.
<i>Zdarzenie</i>	Zaobserwowanie pomyślnie wykonanego zadania.
<i>wyzwalające:</i>	
<i>Przewidywana reakcja:</i>	Sprawdzenie czy jest do dyspozycji osoba oraz wystarczająco materiałów, aby wykonać kolejne zadanie, które jest zależne od właśnie zrealizowanego. Jeżeli zgromadzone materiały są wystarczające oraz istnieje pracownik niewykonujący zadania, wówczas zadanie przechodzi w stan „wykonywane”. Jeśli nie są spełnione powyższe warunki to zadanie przechodzi w stan „zatrzymane”.

EKS020

<i>Krótki opis:</i>	Stan magazynu.
<i>Zdarzenie</i>	Interwał czasowy – liczba godzin np. 24.
<i>wyzwalające:</i>	
<i>Przewidywana reakcja:</i>	Sporządzane jest zestawienie materiałów, których brakuje w magazynie, a następnie jest ono dostarczane kierownikowi.

EKS021

<i>Krótki opis:</i>	Stan zamówienia.
<i>Zdarzenie</i>	Interwał czasowy – pewien ustalony czas.
<i>wyzwalające:</i>	
<i>Przewidywana reakcja:</i>	Sporządzana jest informacja dotycząca zamówienia. Określa ona liczbę zadań, które należy wykonać, aby zrealizować zamówienie. Informacja ta dostarczana jest kierownikowi.

7. PORÓWNANIE BAZ DANYCH ORACLE I MICROSOFT

W niniejszym rozdziale przedstawiono wnioski oraz spostrzeżenia płynące z analizy możliwości tworzenia aktywnych baz danych w SZBD Oracle 11g i Microsoft SQL Server 2005. Informacje te zostały pogrupowane na dwa podrozdziały, zawierające wnioski porównawcze oraz wnioski wynikłe z przeprowadzonych eksperymentów.

7.1. WNIOSKI Z ANALIZY PORÓWNAWCZEJ

W rozdziale 2 zaprezentowano pięć grup zdarzeń prostych. Porównanie możliwości wychwytywania tych zdarzeń w SZBD Oracle 11g i Microsoft SQL Server 2005 jest pierwszym etapem analizy. Drugim etapem są możliwości definiowania w obu SZBD, opisanych w rozdziale 5, operatorów zdarzeniowych. Trzecim etapem są subiektywne wnioski oceniające jakość oraz komfort pracy obu SZBD w zakresie tworzenia aktywnych baz danych.

Analiza rozpoczyna się od porównania możliwości wychwytywania zdarzeń prostych w obu testowanych SZBD.

Pierwszą grupą zdarzeń prostych są zdarzenia związane ze stanem obiektu (CREATE, DROP, ALTER) oraz zdarzenia dostępu (READ, ACCESS). W SZBD Oracle istnieje możliwość definiowania wyzwalacza typu DDL, który pozwala zaobserwować tego typu zdarzenia. Zdarzenia te odnoszą się do wszystkich obiektów i nie ma możliwości skonstruowania wyzwalacza reagującego tylko na jeden typ obiektu. W momencie zaobserwowania takiego zdarzenia, dzięki zmiennym systemowym: ora_sysevent, ora_dict_obj_owner, ora_dict_obj_name możliwe jest uzyskanie informacji o rodzaju zdarzenia, właścicielu oraz nazwie obiektu, którego stan uległ zmianie. Nie istnieją natomiast mechanizmy umożliwiające wychwycenie zdarzeń dostępu do obiektu. Analogiczna sytuacja występuje w SZBD Microsoft SQL Server, jednak przy definiowaniu wyzwalacza aktywowanego takimi zdarzeniami należy określić, które obiekty zostaną poddane obserwacji. W przypadku, gdy zajdzie potrzeba obserwacji omawianych zdarzeń na wszystkich obiektach, należy w definicji wyzwalacza umieścić szczegółową listę zawierającą nazwy tych zdarzeń. Informacje dotyczące nazwy obiektu oraz rodzaju zmiany stanu można uzyskać z obiektu xml, zwróconego przez funkcję „eventdata”. W tabeli 1 zaprezentowano zestawienie możliwości wychwytywania zdarzeń związanych ze stanem obiektu.

Tabela 1. Zdarzenia proste związane ze stanem obiektu

Zdarzenie	Implementacja	
	Oracle	Microsoft
CREATE	✓	✓
DROP	✓	✓
ALTER	✓	✓
ACCESS	✗	✗
READ	✗	✗

✓ – tak

✗ – nie

Drugą grupą zdarzeń prostych są zdarzenia związane ze zmianą stanu transakcji (TBEGIN, TCOMPLETE, TCOMMIT, TROLLBACK). Zarówno w SZBD Oracle jak i w SZBD Microsoft nie istnieje możliwość zdefiniowania wyzwalacza reagującego na tego typu zdarzenia. W tabeli 2 przedstawiono zestawienie możliwości wychwytywania zdarzeń związanych ze stanem transakcji.

Tabela 2. Zdarzenia proste związane ze stanem transakcji

Zdarzenie	Implementacja	
	Oracle	Microsoft
TBEGIN	×	×
TCOMPLETE	×	×
TCOMMIT	×	×
TROLLBACK	×	×

× – nie

Trzecią grupą zdarzeń prostych są zdarzenia temporalne (AT TIME, AFTER TIME, EVERY TIME). W SZBD Oracle dostępne są dwa pakiety zawierające procedury, które odpowiedzialne są za konstruowanie oraz zarządzanie zdarzeniami czasowymi (DBMS_JOB, DBMS_SCHEDULER). Przy tworzeniu tych zdarzeń wymagane jest również podanie częstotliwości interwału czasowego (oprócz zdarzenia AT TIME wywołującego się tylko raz w określonym czasie). Zdarzenie wywołujące się po pewnym czasie (AFTER TIME) należy skonstruować tak, aby data uruchomienia zdarzenia była opóźniona o zadany interwał czasowy lub tworzone zdarzenie było nieaktywne. Jeżeli jest ono nieaktywne, wówczas należy je aktywować przed upływem zadanego interwału czasowego. Istnieje także możliwość definiowania zdarzeń wykonujących się periodycznie (EVERY TIME). Dzięki rozbudowanej semantyce interwału czasowego możliwe jest bardzo dokładne określenie odstępu czasu pomiędzy kolejnymi zdarzeniami. W SZBD Microsoft SQL Server również istnieje możliwość definiowania, wyżej wymienionych, zdarzeń temporalnych, ale konstruowanych przy użyciu innych mechanizmów. Interwał

Tabela 3. Zdarzenia proste temporalne

Zdarzenie	Implementacja	
	Oracle	Microsoft*
AT TIME	✓	✓
AFTER TIME	✓	✓
EVERY TIME	✓	✓

* z wyjątkiem środowiska darmowego SQL SERVER EXPRESS

✓ – tak

czasowy może wynieść minimalnie jedną minutę, a nie tak jak w SZBD Oracle nawet jedną sekundę. Istnieje jednak możliwość przyłączenia wielu harmonogramów do jednego zdarzenia czasowego, przesuniętych względem siebie o zadany interwał czasowy, wyrażony w sekundach. Dzięki temu możliwe jest ominięcie ograniczenia interwału i uzyskanie sekundowej częstotliwości wywoływania zdarzenia. Oba SZBD umożliwiają definiowanie w zdarzeniach czasowych akcji, której skutkiem jest uruchomienie aplikacji zainstalowanej w systemie operacyjnym. W tabeli 3 pokazano zestawienie możliwości tworzenia zdarzeń temporalnych.

W czwartej grupie zdarzeń prostych znajdują się zdarzenia aktywowane przez manipulacje na krotkach relacji i perspektywy (INSERT, UPDATE, DELETE). Zarówno SZBD Oracle jak i SZBD Microsoft SQL Server wspierają w pełni powyższe zdarzenia. W tabeli 4 zaprezentowano zestawienie możliwości wychwytywania zdarzeń manipulacji na krotkach.

Tabela 4. Zdarzenia manipulacji na krotkach

Zdarzenie	Implementacja	
	Oracle	Microsoft
INSERT	✓	✓
UPDATE	✓	✓
DELETE	✓	✓

✓ – tak

Piąta grupa zdarzeń prostych zawiera zdarzenia LOGIN, LOGOUT, STARTUP, SHUTDOWN, ERROR. W SZBD Oracle istnieje możliwość zaobserwowania zdarzeń autoryzacji (LOGIN) do bazy danych oraz zdarzeń odłączania (LOGOUT) od bazy danych. Występuje również możliwość zdefiniowania wyzwalacza reagującego na zdarzenia włączania (STARTUP) i wyłączenia (SHUTDOWN) bazy danych oraz na zdarzenia wystąpienia błędu (ERROR). W SZBD Microsoft SQL Server dozwolone jest wyłącznie wychwycenie zdarzenia autoryzacji (LOGIN). W tabeli 5 przedstawiono zestawienie możliwości wychwytywania zdarzeń występujących w SZBD.

Tabela 5. Zdarzenia występujące w SZBD

Zdarzenie	Implementacja	
	Oracle	Microsoft
LOGIN	✓	✓ *
LOGOUT	✓	×
STARTUP	✓	×
SHUTDOWN	✓	×
ERROR	✓	×

* dostępny dopiero z SP2

✓ – tak

× – nie

Drugim etapem analizy porównawczej SZBD Oracle i Microsoft SQL Server są możliwości definiowania operatorów zdarzeniowych w obu SZBD. W oparciu o te operatory budowane są zdarzenia złożone. W obu SZBD do utworzenia operatora zdarzeniowego potrzebna jest pomocnicza relacja (w Oracle może to być też pakiet) w której przechowana jest informacja o wystąpieniach zdarzeń elementarnych.

Pierwszą grupą operatorów zdarzeniowych są operatory logiczne (AND, OR, NOT). Za ich pomocą mogą być tworzone wyrażenia, które w przypadku spełnienia powodują wywołanie zdarzenia złożonego. W SZBD Oracle oraz w SZBD Microsoft SQL Server występują mechanizmy umożliwiające konstruowanie takich zdarzeń. Cała logika wyrażenia zawarta jest w specjalnie do tego celu zdefiniowanym wyzwalaczu. W tabeli 6 pokazano możliwości konstruowania operatorów logicznych.

Tabela 6. Operatory logiczne

Operator	Implementacja	
	Oracle	Microsoft
AND	✓	✓
OR	✓	✓
NOT	✓	✓

✓ – tak

Drugą grupą operatorów zdarzeniowych są operatory następstwa (sequence, prior, relative, PLUS). Operatory te charakteryzują się tym, że określają kolejność występowania zdarzeń elementarnych, na podstawie których są skonstruowane. W SZBD Oracle istnieje możliwość zdefiniowania powyższych operatorów, jednak ze względu na ograniczenia wyzwalaczy, w ciele wyzwalacza nie można utworzyć opóźnionego zdarzenia czasowego. W związku z tym, nie ma możliwości utworzenia operatora PLUS. Każdy operator realizowany jest w postaci wyzwalacza wykonującego całą logikę. W SZBD Microsoft SQL Server możliwe jest zdefiniowanie wszystkich operatorów, ponieważ nie występują ograniczenia wyzwalaczy typowe dla Oracle. W tabeli 7 zaprezentowano możliwości konstruowania operatorów następstwa.

Tabela 7. Operatory następstwa

Operator	Implementacja	
	Oracle	Microsoft
sequence	✓	✓
prior	✓	✓
relative	✓	✓
PLUS	x	✓

✓ – tak

x – nie

Trzecią grupą operatorów zdarzeniowych są operatory wyliczeniowe (cykliczny *every*, niecykliczny). Dzięki nim możliwe jest określenie częstotliwości występowania zdarzenia złożonego. W przypadku operatora niecyklicznego zdarzenie złożone może wystąpić co najwyżej raz, natomiast przy użyciu operatora cyklicznego *every* wielokrotnie. Zarówno w SZBD Oracle, jak i w SZBD Microsoft SQL Server istnieje możliwość definiowania wyzwalaczy pełniących role poszczególnych operatorów. W tabeli 8 przedstawiono możliwości konstruowania operatorów wyliczeniowych.

Tabela 8. Operatory wyliczeniowe

Operator	Implementacja	
	Oracle	Microsoft
cykliczny <i>every</i>	✓	✓
niecykliczny	✓	✓

✓ – tak

Operator okna (*firstAfter*) tworzy logiczne okno, w obrębie którego wystąpienie zdefiniowanego zdarzenia elementarnego powoduje wywołanie zdarzenia złożonego. W obu SZBD istnieje możliwość zdefiniowania wyzwalacza pełniącego rolę operatora okna (tabela 9).

Operator aperiodyczny może zostać zdefiniowany jako operator z kumulacją lub bez kumulacji. W przypadku operatora z kumulacją zdarzenie złożone zostanie wywołane tylko jeden raz, w momencie wystąpienia zdarzenia kończącego, po uprzednim zaobserwowaniu zdarzenia pośredniczącego. Natomiast operator bez kumulacji wywoła zdarzenie złożone za każdym razem, gdy wystąpi zdarzenie pośredniczące. Zarówno w SZBD Oracle jak i w SZBD Microsoft SQL Server istnieje możliwość zdefiniowania obu rodzajów operatorów aperiodycznych (tabela 9).

Operator periodyczny również posiada dwa warianty: z kumulacją i bez kumulacji. W przypadku operatora z kumulacją zdarzenie złożone zostanie wywołane tylko jeden raz, po upływie określonego czasu. Natomiast operator bez kumulacji wywoła zdarzenie złożone w sposób cykliczny, co pewien zadany interwał czasowy. SZBD Microsoft SQL Server wspiera definiowanie tego typu operatorów, natomiast w SZBD Oracle, ze względu na ograniczenia wyzwalaczy, nie można utworzyć zdarzenia czasowego w ciele wyzwalacza. W związku z tym, nie ma możliwości konstruowania operatora periodycznego. W tabeli 9 pokazano możliwości konstruowania operatorów okna, aperiodycznych oraz periodycznych.

Trzecim etapem analizy porównawczej SZBD Oracle i Microsoft SQL Server są subiektywne wnioski oceniające jakość oraz komfort pracy obu SZBD w zakresie tworzenia aktywnych baz danych.

Tabela 9. Operator okna *firstAfter*, aperiodyczny, periodyczny

Operator	Implementacja	
	Oracle	Microsoft
okna (<i>firstAfter</i>)	✓	✓
aperiodyczny z kumulacją	✓	✓
aperiodyczny bez kumulacji	✓	✓
periodyczny z kumulacją	✗	✓
periodyczny bez kumulacji	✗	✓

✓ – tak

✗ – nie

Narzędzia komunikacyjne dostarczone wraz z SZBD Microsoft SQL Server umożliwiają większy komfort i wygodę pracy. Oprócz standardowej komunikacji za pomocą komend tekstowych, dostarczony jest kreator pozwalający na definiowanie obiektów. Ułatwia to posługiwanie się strukturami języka SQL. W narzędziu dostarczonym przez Oracle nie ma możliwości utworzenia zdarzenia czasowego za pomocą kreatora. W SZBD Microsoft SQL Server zdarzenia czasowe wykonywane są przez usługę *SQLServerAgent*, zatem należy ją włączyć, aby zdarzenia czasowe były wykonywane. Narzędzie firmy Microsoft posiada bardziej przyjazny interfejs graficzny. Zmienne w języku dedykowanym SQL Server (T-SQL) posiadają prefiks „@”, co wyraźnie odróżnia je od innych ciągów znakowych. Zarówno w języku T-SQL (Microsoft), jak i w PL/SQL (Oracle) występują instrukcje sterujące (WHILE, FOR, IF itp.), ułatwiające konstruowania bloków anonimowych, procedur, funkcji oraz wyzwalaczy. SZBD Oracle zapewnia większą zgodność ze standardem SQL-92 oraz SQL-99. Komunikaty informujące o wystąpieniu błędu są bardziej szczegółowe i dokładniejsze dla SZBD Microsoft SQL Server. W tabeli 10 zaprezentowano ocenę porównawczą cech obu analizowanych baz danych.

Tabela 10. Ocena cech środowisk oraz narzędzi Oracle i Microsoft

Cecha	Implementacja	
	Oracle	Microsoft
Interfejs graficzny	●●●●○	●●●●●
Wygodę użytkowania	●●●○○	●●●●○
Zalety języka	●●●●○	●●●●○
Wydajność bazy danych	●●●●●	●●●●●
Informacja o błędzie	●●●●○	●●●●○
Możliwości interfejsu	●●●●○	●●●●●

●●●●● – bardzo wysoki

●●●●○ – wysoki

●●●○○ – średni

●●○○○ – niski

●○○○○ – bardzo niski

W SZBD Microsoft SQL Server przydatna byłaby możliwość rejestrowania zdarzeń odłączania (LOGOUT) od bazy danych oraz wychwycenia zdarzenia włączenia (STARTUP) i wyłączenia (SHUTDOWN) bazy danych. Podyktowane jest to częstą potrzebą wykonania określonych czynności powiązanych z tymi zdarzeniami. SZBD Oracle nie umożliwia dynamicznego tworzenia zdarzeń czasowych, co przekłada się na mniejszą aktywność bazy danych. Aktywności takie bywają potrzebne np. gdy w momencie wykonywania zadania, tworzone jest zdarzenie czasowe, które określa czas na realizację tego zadania. Brak wykonania zadania powoduje wówczas wywołanie akcji zdefiniowanej w zdarzeniu złożonym, informując o opóźnieniu.

7.2. WNIOSKI Z PRZEPROWADZONYCH EKSPERYMENTÓW

Zdefiniowane w rozdziale 6 eksperymenty dostarczyły szeregu wniosków dotyczących możliwości wykorzystania aktywności w SZBD Oracle 11g i Microsoft SQL Server 2005. Eksperyment (EKS001), który miał na celu zaobserwowanie reakcji aktywnej bazy danych w momencie stwierdzenia braku potrzebnych materiałów do wykonania zadania, zakończył się powodzeniem. Oba SZBD, dzięki posiadanym mechanizmom, umożliwiły implementację takiej aktywności bazy danych. Należy jednak zaznaczyć, że w SZBD Oracle nie ma możliwości, w wyzwalaczu wierszowym, manipulacji na krotkach relacji, dla której został on utworzony. Dlatego przy eksperymentach należało skonstruować dwa wyzwalacze: wierszowy i poleceniowy, które wywoływane były jednym zdarzeniem prostym. Wyzwalacz wierszowy był odpowiedzialny za gromadzenie w zmiennych pakietu informacji o dokonanych zmianach w relacji, a wyzwalacz poleceniowy, korzystając z tych informacji, wykonywał całą wymaganą logikę. W SZBD Microsoft SQL Server istnieje możliwość skonstruowania wyzwalacza, w którego „ciele” wykonywane będą operacje go wywołujące. Ograniczeniem takich wyzwalaczy jest poziom rekurencyjnego wywołania, który wynosi 32, więc jeżeli automatyczny proces będzie zbyt długi, wówczas należy posłużyć się dodatkową relacją. Kolejny eksperyment (EKS002) pokazał, że w obu SZBD jest możliwa archiwizacja danych personalnych w momencie manipulacji na krotkach relacji opisującej osoby. Proces przemysłowy został zatrzymany, gdy zabrakło materiałów, natomiast miał on być wznowiony (EKS003), gdy potrzebne materiały zostały dostarczone do magazynu (EKS010). Zarówno SZBD Oracle jak i Microsoft SQL Server pozwalają zamodelować takie sytuacje. Problemu również nie stanowi realizacja przydzielenia kolejnego zadania (EKS007), wykonanie kolejnego zadania (EKS004) oraz zrealizowanie określonego zadania (EKS008). Są to zdarzenia proste powiązane między sobą, tworzące w ten sposób cykl zależności. Wymienione eksperymenty w obu SZBD zakończyły się powodzeniem. Dużym problemem okazało się konstruowanie zdarzeń czasowych, ponieważ w SZBD Oracle nie ma możliwości dynamicznego tworzenia zdarzenia czasowego w „ciele” wyzwalacza. Eksperymenty EKS005, EKS006 oraz

EKS012, miały na celu poinformowanie kierownika o opóźnieniach przy realizacji zadania, zamówienia oraz dostawy materiałów do magazynu. Eksperymenty te realizuje operator następstwa PLUS. W SZBD Microsoft SQL Server w momencie rozpoczęcia któregoś z wyżej wymienionych zdarzeń, definiowano dodatkowo zdarzenie czasowe, które zostało wykonane raz po upływie czasu, jaki był zarezerwowany na realizację. Zdarzenia były tworzone w sposób dynamiczny w „ciele” wyzwalacza i usuwane, gdy poinformowały o opóźnieniu lub, gdy realizacja zakończyła się w przewidywanym czasie. W SZBD Oracle nie ma możliwości konstruowania i manipulacji na zdarzeniach czasowych w „ciele” wyzwalacza. Rozwiązaniem tego problemu mogłoby być niezależne zdarzenie czasowe, wykonujące się z określoną częstotliwością, które sprawdzałoby czy jakieś zadanie, zamówienie lub dostawa materiałów nie jest opóźniona. Przy dużej częstotliwości rozwiązanie to będzie zbyt mocno obciążać bazę danych, natomiast przy zbyt małej częstotliwości nie wszystkie opóźnienia zostaną zaobserwowane i odnotowane. Eksperymenty rozpoczęcia realizacji zamówienia (EKS013), zakończenia realizacji zamówienia (EKS009), zatwierdzenia realizacji zamówienia (EKS014) oraz zamówienia potrzebnych materiałów (EKS011) testują zdarzenia proste. Tak samo, jak poprzednio zdarzenia proste nieczasowe w łatwy sposób pozwoliły się zamodelować zarówno w SZBD Oracle jak i w SZBD Microsoft SQL Server. Realizacją operatora wyliczeniowego niecyklicznego był eksperyment EKS015. Piąte zdarzenie nie dostarczenia na czas materiałów przez firmę, skutkowało poinformowaniem kierownika i zasugerowaniem mu, aby zmienić dostawcę materiałów. Do zamodelowania takiego operatora konieczna była dodatkowa relacja, przechowująca informację o liczbie przypadków opóźnień dostaw materiałów dla każdego z dostawców. Ze względu na to, że w SZBD Oracle nie ma możliwości skonstruowania zdarzenia czasowego w sposób dynamiczny, realizacja tego eksperymentu była jedynie przybliżona. W SZBD SQL Server istnieją odpowiednie mechanizmy realizacji operatora PLUS. Eksperyment EKS016 miał na celu wytypowanie pracowników, którzy notorycznie opóźniają realizację zadań. Informacja o pracowniku, który trzy razy z kolei opóźnił wykonanie zadania, została dostarczona kierownikowi. Realizacją tego zdarzenia jest wyrażenie logiczne składające się z dwóch operatorów następstwa, sekwencji oraz operatora logicznego AND. Operatory te oparte były jednak na zdarzeniu czasowym, tworzonym w sposób dynamiczny. Wynika z tego, że eksperyment został zrealizowany w pełni w SZBD Microsoft SQL Server, a w SZBD Oracle jego realizacja była jedynie przybliżona. Do przeprowadzenia tego eksperymentu została wykorzystana dodatkowa relacja, umożliwiająca dla każdego pracownika przechowywanie liczby opóźnień wykonywanych zadań. Eksperyment EKS017 polegający za zmniejszeniu kwoty zamówienia o 3% w przypadku opóźnienia realizacji, w SZBD Microsoft SQL Server został wykonany w pełni. Jest to zdarzenie czasowe, więc tak jak wspomniano powyżej, tego typu zdarzenia, przy braku możliwości dynamicznego ich konstruowania w SZBD Oracle, dostarczają problemów, które można rozwiązać jedynie w sposób przybliżony.

Innym rodzajem eksperymentów są eksperymenty, których akcja wykonywana jest automatycznie. W przypadku realizacji eksperymentu EKS019, czyli wykonywania kolejnych zadań przez specjalistyczną maszynę zarówno w SZBD Oracle, jak i w SZBD Microsoft SQL Server notowane są pewne ograniczenia. Odpowiednia akcja jest oparta na relacji przechowującej zadania do wykonania, co w SZBD Oracle wymaga zdefiniowania dwóch wyzwalaczy. W SZBD Microsoft SQL Server natomiast istnieje ograniczenie 32 wzajemnych wywołań wyzwalaczy, co w przypadku nawet niedługiego procesu technologicznego może zostać łatwo przekroczone. Celem eksperymentu (EKS018) była archiwizacja krotek z relacji zamówienia przy wyłączaniu bazy danych. Jest to zdarzenie proste, lecz SZBD Microsoft SQL Server go nie wspiera. Możliwe jest zdefiniowanie zdarzenia czasowego, które codziennie o określonej godzinie, przed wyłączeniem bazy danych, będzie przeprowadzało archiwizację. Takie rozwiązanie jest dobre wtedy, gdy czas zamykania bazy jest znany. SZBD Oracle

Tabela 11. Eksperymenty w aktywnej bazie danych

Eksperyment	Implementacja	
	Oracle	Microsoft
EKS001	✓	✓
EKS002	✓	✓
EKS003	✓	✓
EKS004	✓	✓
EKS005	x ^	✓
EKS006	x ^	✓
EKS007	✓	✓
EKS008	✓	✓
EKS009	✓	✓
EKS010	✓	✓
EKS011	✓	✓
EKS012	x ^	✓
EKS013	✓	✓
EKS014	✓	✓
EKS015	x ^	✓
EKS016	x ^	✓
EKS017	x ^	✓
EKS018	✓	x ^
EKS019	✓	✓*
EKS020	✓	✓
EKS021	✓	✓
EKS022	✓	✓

* 32 poziomy zagnieżdżenia wyzwalaczy

^ próby ominięcia ograniczenia

✓ – możliwe

x – niemożliwe

posiada mechanizmy, dzięki którym możliwe jest zamodelowanie tej aktywności. Eksperymenty EKS020, EKS021 oraz EKS022, cyklicznie co pewien ustalony czas informowały, jaki był stan magazynowy, ile zamówień zostało zrealizowanych oraz ile rozdzielnic zostało skonstruowanych. Są to zdarzenia czasowe periodyczne i nie istnieje potrzeba ich tworzenia w wyzwalaczu, dlatego w obu SZBD można zamodelować takie aktywności. W tabeli 11 przedstawiono informacje dotyczące realizacji poszczególnych eksperymentów.

8. PODSUMOWANIE

Zarówno SZBD Oracle 11g, jak i SZBD Microsoft SQL Server 2005 dostarczają mechanizmów, za pomocą których można skonstruować aktywną bazę danych. Zdarzenia proste zostały podzielone na pięć grup, które zawierają zdarzenia manipulacji na obiektach oraz krotkach relacji, a także zdarzenia związane z czasem, określające stan transakcji, czy też związane z SZBD. W obu SZBD nie ma możliwości wychwycenia zdarzeń dostępu oraz odczytu stanu obiektu. Nie istnieją mechanizmy, które by na to pozwalały. Microsoft oraz Oracle nie wspierają również zdarzeń związanych ze stanem transakcji. W związku z tym, nie ma możliwości zareagowania np. w momencie zatwierdzenia, wycofania czy też przerwania transakcji. Zarówno w jednym, jak i w drugim SZBD (poza darmową wersją oprogramowania SQL Express firmy Microsoft) występują mechanizmy pozwalające na konstruowanie zdarzeń czasowych. Dzięki nim możliwe jest wykonywanie zdefiniowanych akcji w określonym czasie lub w sposób periodyczny. Mogą one być wykorzystywane między innymi w procesach technologicznych, czy też procesach kontrolujących prace określonych urządzeń. SZBD Microsoft SQL Server i SZBD Oracle wspierają w pełni wychwytywanie zdarzeń manipulacji na krotkach relacji lub perspektywy. Dzięki temu istnieje możliwość zareagowania na manipulację przez wykonanie określonych akcji. Mechanizmy te stają się bardzo przydatne w momencie, gdy zachodzi potrzeba przechowywania danych archiwalnych. Jak wiadomo, baza danych zawiera aktualny stan danych, więc bez archiwizacji wszelkiego rodzaju informacje o zmianach, byłyby bezpowrotnie tracone. SZBD Oracle dostarcza mechanizmy pozwalające na wychwytywanie zdarzeń związanych z SZBD, natomiast SZBD Microsoft SQL Server (z SP2) umożliwia jedynie zaobserwowanie zdarzenia autoryzacji do serwera bazy danych. Wykonywanie określonych akcji w momencie uruchamiania i zatrzymywania bazy danych jest sprawą kluczową. Tymczasem w SZBD Microsoft SQL Server nie ma możliwości zareagowania na tego typu zdarzenia. Często w momencie odłączania użytkownika od bazy danych wykonywane są określone akcje np. „czyszczenie” relacji tymczasowych, zdarzenie to nie jest jednak wspierane przez oprogramowanie Microsoft.

Do budowy zdarzeń złożonych wykorzystywane są operatory zdarzeniowe oraz zdarzenia elementarne. Operatory zgromadzone w grupie zdarzeń logicznych, pozwalają na definiowanie logicznych wyrażeń zdarzeniowych. Dzięki nim możliwe jest

zapisanie skomplikowanego, złożonego z więcej niż jednego, zdarzenia warunku logicznego. Zarówno SZBD Oracle, jak i SZBD Microsoft SQL Server posiadają mechanizmy, za pomocą których możliwe jest konstruowanie tego typu wyrażenia logicznego. W kolejnej grupie występują operatory następstwa, których najistotniejszą cechą jest porządkowanie zdarzeń na osi czasu. Operatory te można wykorzystać w sytuacji, gdy specyficzna kolejność występowania zdarzeń elementarnych, powoduje zaistnienie zdarzenia. W obu SZBD występują odpowiednie mechanizmy, pozwalające na zdefiniowanie tego typu operatorów. Wyjątkiem jest operator PLUS, który w SZBD Oracle wymaga użycia innych mechanizmów. W następnej grupie znalazły się operatory wyliczeniowe, określające, które bądź co które zdarzenie elementarne ma spowodować spełnienie wyrażenia opartego na tego typu operatorze. Można je wykorzystać np. w dziale sprzedaży, określając, że co tysięczny klient otrzymuje rabat na zakupy. Zarówno SZBD Oracle, jak i SZBD Microsoft SQL Server posiadają mechanizmy umożliwiające zdefiniowanie tych operatorów. W ostatniej grupie znajdują się operatory, które tworzą logiczne okna, wewnątrz których wystąpienie określonego zdarzenia, powoduje pomyślną weryfikację wyrażenia zdarzeniowego, na którym są one oparte. Operatory te mogą być użyteczne przy obserwowaniu nieporządkanych zdarzeń pomiędzy konkretnymi zdarzeniami. SZBD Oracle i Microsoft SQL Server dostarczają odpowiednich mechanizmów, za pomocą których możliwe jest skonstruowanie tego typu operatorów. Wyjątek stanowi operator *periodyczny*, którego realizacja w SZBD Oracle wymaga zastosowania innych mechanizmów.

Przeprowadzone eksperymenty na aktywnej bazie danych pokazały, że oba SZBD posiadają skuteczne mechanizmy realizowania zdarzeń prostych. SZBD Oracle posiada jednak dużą wadę, ponieważ nie umożliwia konstruowania prostych zdarzeń czasowych w sposób dynamiczny. Jest to ważny aspekt, ponieważ często istnieje potrzeba korzystania z takiej funkcjonalności. SZBD Microsoft SQL Server natomiast nie posiada możliwości konstruowania kilku rodzajów zdarzeń prostych, co wpływa na zmniejszenie liczby zdarzeń, na które może reagować. Oba SZBD pozwalają na modelowanie operatorów zdarzeniowych, jednak w SZBD Oracle podczas konstrukcji operatorów czasowych należy użyć innych dostępnych mechanizmów, przybliżających określoną funkcjonalność. Zdarzenia czasowe nie tworzone w sposób dynamiczny dostępne są zarówno w SZBD Oracle, jak i w SZBD Microsoft SQL Server.

Narzędzia dostarczone przez Microsoft posiadają bardziej szczegółowe i dokładniejsze komunikaty informujące o wystąpieniu błędu, co sprawia, że czas potrzebny na zlokalizowanie błędu jest krótszy. W narzędziach udostępnionych przez Oracle nie ma możliwości utworzenia, za pomocą kreatora, zdarzenia czasowego. Oprogramowanie w obu systemach bazodanowych posiada struktury sterujące w dedykowanych językach, co pozwala na rozszerzenie możliwości języka SQL. Narzędzie Microsoft posiada bardziej przyjazny i bardziej ergonomiczny interfejs do interakcji z użytkownikiem.

Reasumując, mimo drobnych braków, oba porównywane SZBD dostarczają skutecznych mechanizmów do tworzenia aktywnych baz danych.

SZBDCR powstały w efekcie próby połączenia, z zachowaniem wszystkich zalet, tradycyjnych SZBD (Systemów zarządzania bazami danych) i technologii czasu rzeczywistego.

Ze względu na swoje własności (szczególnie na fakt, iż czas wykonania transakcji jest przewidywalny i obliczalny), bazy danych czasu rzeczywistego znajdują wiele zastosowań w dziedzinach takich jak: bankowość, aktualizacje giełdowe, śledzenie radarowe, telekomunikacja, teleinformatyka, kontroling przetwarzania, systemy kontroli floty pojazdów, systemy multimedialne, systemy rejestracyjne, systemy analizujące dane i prowadzące statystyki.

2. OGRANICZENIA CZASOWE W SZBDCR

System zarządzania bazą danych czasu rzeczywistego w swojej podstawowej funkcjonalności nie różni się od tradycyjnego SZBD i powinien umożliwić:

- zapisywanie danych,
- przechowywanie danych,
- wyszukiwanie danych,
- usuwanie danych,
- aktualizowanie danych.

Istotna różnica pojawia się w związku z ograniczeniami czasu rzeczywistego. Pewne zadania i procesy systemu zobligowane są do przestrzegania rygorystycznych, z góry narzuconych, ograniczeń czasowych. Stopień tych ograniczeń jest ściśle związany z konkretnym zastosowaniem. Czasem wyszukiwane informacje lub przetwarzane dane są tak istotne, że nie dopuszcza się możliwości przekroczenia limitów czasowych. W innym zastosowaniu może okazać się, że wartość lub użyteczność danych stopniowo spada po przekroczeniu oczekiwanego czasu, jednakże ważniejsze są jakkolwiek spóźnione dane aniżeli brak tych danych. W tym przypadku pojawiają się inne metody sterowania przetwarzaniem transakcji – opóźniona transakcja może uzyskać największy priorytet, tak aby jak najszybciej ukończyć swoje zadania lub zupełnie odwrotnie – może zostać przesunięta na koniec kolejki priorytetowej określającej kolejność przetwarzania. W ostateczności może zostać przerwana z uwagi na to, że jej dalsze wykonywanie prawdopodobnie istotnie wpłynie na opóźnienie innych transakcji wykonywanych w tym samym czasie. Głównie z uwagi na wysoką korelację pomiędzy transakcjami i konieczność współdzielenia zasobów (przy założeniu braku możliwości przetwarzania rozproszonego bądź równoległego przetwarzania macierzowego (ang. *grid*)).

Transakcje, o których mowa w poprzednim akapicie, nazywane są transakcjami czasu rzeczywistego. Z kolei SZBDCR, które przyzwalają na wykonanie lub dokończenie transakcji po upływie czasu przeznaczanego na jej działanie nazywane są „częściowo czasu rzeczywistego” (ang. *Soft Real-Time*) w odróżnieniu od tych, które wyłączają niezakończone transakcje (ang. *Hard Real-Time*).

3. SZEREGOWANIE ZADAŃ W SZBDCR

Istotną kwestią jest precyzyjne określenie wad i ograniczeń tradycyjnych SZBD, które to powodują, iż nie mogą one sprostać określonym wymaganiom czasowym, implikując tym samym wzmożone badania nad SZBDCR. Przyczyn takiego stanu rzeczy jest kilka:

- Ze względu na aktualne możliwości sprzętowe trudno osiągnąć stały (nieprzekraczający wartości odchylenia standardowego) i skalowalny (o narzucie czasowym proporcjonalnie zmieniającym się wraz z ilością przetwarzanych danych) czas przetwarzania tej samej transakcji wykonanej w różnych momentach działania SZBD.
- Proces przetwarzania transakcji obarczony jest trudną do określenia a priori ilością operacji dyskowych wejścia/wyjścia – głównie ze względu na błędy strony – błąd występujący w sytuacji, kiedy operacja odwołuje się do rekordu na stronie niebędącej aktualnie w *cache*'u (buforze stron). Naturalną konsekwencją takiej sytuacji jest konieczność odczytania wzywanej strony z pliku bazy (wymagana jest operacja dyskowa, która jest kilka rzędów jednostki czasu wolniejsza niż pobieranie strony z bufora stron).
- SZBD posiadają różne protokoły obsługi i szeregowania transakcji na różnym poziomie izolacji (własność transakcji zapewniająca niezależność wykonywanych współbieżnie transakcji) co, w przypadku konfliktu, zakleszczenia lub nieserializowalności (własność planu transakcji gwarantująca, że rezultat transakcji wykonywanych równolegle jest taki sam, jak wykonanie tych transakcji jedna po drugiej), skutkuje koniecznością wykonania operacji wyłączenia i restartowania danej transakcji (odwołania zmian przeprowadzonych do tej pory w ramach jej operacji, zwolnienie blokad na zbiorach danych, w końcu ponowne uruchomienie całego procesu przetwarzania). Stąd też wiele pomysłów na SZBDCR rozważa realizację bazy w pamięci operacyjnej. Takie rozwiązanie niesie ze sobą oczywiste wady. Ciągłe pozostają problemy z ilością dostępnej pamięci i jej ulotnością w przypadku utraty zasilania. Taka propozycja może być więc alternatywą tylko dla stosunkowo małych baz danych. Jednakże wtedy powstaje wątpliwość co do zasadności wprowadzania bazy danych czasu rzeczywistego, skoro tworzona baza danych jest niewielkich rozmiarów i może zostać umieszczona w odpowiednio powiększonym buforze zarządzanym przez menadżera buforów klasycznego SZBD. Zakłada się tutaj brak zapisywania stron na dysku bezpośrednio po modyfikacji rekordu (krotki danych) lub zapisywanie opóźnione w czasie, np. w czasie bezczynności.

Kolejnym, nietrywialnym problemem, który pojawia się w związku z ograniczeniami czasu rzeczywistego, jest szeregowalność zadań. SZBDCR powinien w swojej implementacji zapewniać efektywny algorytm, który przygotowuje i aktualizuje w czasie rzeczywistym kolejkę zadań uwzględniając ograniczone zasoby i, co ważniejsze, ograniczenia czasowe transakcji wraz z ich priorytetami. Nie sposób jednak skorzy-

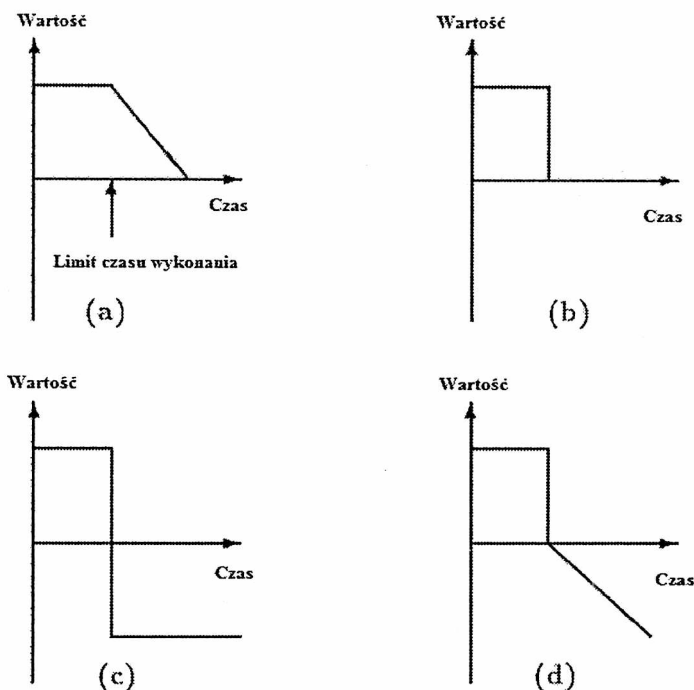
stać tutaj z konwencjonalnych i powszechnie znanych algorytmów służących do szeregowania zadań opisanych przez J.L. Petersona i A. Silberschatza [PetSil 1985]. Zadaniem tych algorytmów jest balansowanie dostępnymi zasobami w taki sposób, aby każde z zadań otrzymało równy udział. Głównie po to, aby zmaksymalizować użyteczność i wydajność systemu – w odróżnieniu do SZBDCR, gdzie zakłada się, że analogiczny algorytm powinien w pierwszej kolejności uwzględniać krytyczność i limit czasowy transakcji (zadania). Takie wytyczne w sposób naturalny implikują sytuację, w której zostanie zachwiana istotnie równość udziału w zasobach poszczególnych zadań i powstanie rozbieżność w stosunku do optymalnego uszeregowania, czyli najbardziej efektywnego wykorzystania systemu.

W SZBDCR transakcje powinny być wykonywane w kolejności priorytetowej – im wyższy priorytet został nadany transakcji tym więcej zasobów zostanie jej przydzielonych na dłuższy okres. Największy problem sprawia prawidłowe wyznaczenie priorytetu każdej z transakcji, tak aby wykonanie zadań w przypisanej kolejności zagwarantowało spełnienie narzuconych ograniczeń czasowych. Ben Kao i H. Garcia-Molina w pracy [GarKao 1995] wyszczególnili podstawowe czynniki (wskazane poniżej), które powinny mieć największy wpływ na wyznaczanie priorytetu transakcjom.

1. **Krytyczność (γ)** – im bardziej istotna dla systemu jest dana transakcja tym większy priorytet powinna uzyskać. Wartość ta może być określona funkcją. Przykładowe funkcje przedstawiono na rysunku 1.
2. **Ograniczenie czasowe (d)** – im bliżej granicy czasu wykonania danej transakcji tym większy priorytet powinna posiadać.
3. **Rozmiar niewykonanej pracy (l)** – im więcej pracy pozostało do wykonania w danej transakcji tym mniejszy priorytet. W sytuacji, gdy transakcja dotarła już do etapu zatwierdzania (ang. *commit phase*), jej priorytet powinien być największy, gdyż etap ten wymaga najmniejszej mocy obliczeniowej, szybko się kończy i powoduje zwolnienie odpowiednich zasobów redukując przy tym prawdopodobieństwo wystąpienia blokady.
4. **Wielkość obliczeń zrealizowanych w ramach danej transakcji (c)** – transakcje, które wykonały więcej obliczeń niż inne, powinny mieć większy priorytet z uwagi na to, że ich zakończenie zwolni większą ilość zasobów w krótszym czasie. Dodatkowo, koszt przeprowadzenia operacji wycofywania (ang. *rollback*) jest bardziej kosztowny dla bardziej zaawansowanych w działaniu transakcji.
5. **Wiek (a)** – transakcje, które rozpoczęły się wcześniej (starsze) powinny mieć większy priorytet niż późniejsze (młodsze). Pozwala to na utrzymanie spójności w danych zewnętrznych.
6. **Bufor czasowy (s)** – określa ilość czasu, podczas którego dana transakcja nie będzie wykonywana, a mimo wszystko zdąży wykonać się jeszcze na czas. Jeśli za t_a przyjąć czas zgłoszenia transakcji, to bufor czasowy (s) można wyrazić wzorem:

$$s = d - t_a - c - l$$

Naturalnie im mniejszy bufor czasowy, tym większy priorytet powinien zostać nadany transakcji.



Rys. 1. Przykładowe wykresy funkcji określającej krytyczność transakcji.
Opóźnionej transakcji przypisana jest: (a) wartość dodatnia malejąca, (b) wartość zero, (c) wartość ujemna, (d) wartość ujemna malejąca. Źródło [GarKao 1995]

W rzeczywistości, sposób doboru kombinacji powyższych czynników nie jest trywialny i w dużej mierze zależy od konkretnego zastosowania. Propozycja ogólnego wzoru na określenie priorytetu transakcji jest następująca [ShRaLe 1988]:

$$pr(T) = \gamma(w_1a - w_2d + w_3c - w_4l)$$

gdzie:

$pr(T)$ – priorytet transakcji T ,
 w_i – wagi istotności czynników.

Najprawdopodobniej, w ogólności, kształt wzoru i dobór wag jest silnie skorelowany z przyjętą formą implementacji, zastosowanymi technologiami i środowiskiem.

Następną, istotną kwestią, na którą mają wpływ ograniczenia transakcji czasu rzeczywistego jest kontrola współbieżności. Konwencjonalne systemy bazodanowe korzystają w większości z popularnego i sprawdzonego w zastosowaniach komercyjnych protokołu dwufazowego blokowania (ang. *2PL – Two-Phase Locking*). Polega on na wyznaczeniu dwóch etapów w transakcji:

- pierwszy, w którym potrzebne zasoby są rezerwowane lecz nie mogą być zwalniane
- drugi, w którym wszystkie zarezerwowane zasoby i założone blokady są zwalniane bez możliwości rezerwowania kolejnych.

Takie rozwiązanie jest niesatysfakcjonujące w SZBDCR, gdzie szeregowanie zadań powinno być oparte o priorytety transakcji (a nie, jak w przypadku konwencjonalnych SZBD, podyktowane optymalnym układem transakcji ekwiwalentnym do serializowalnego wykonania ich w sekwencji). Konwencjonalny algorytm będzie w efekcie generował częste zakleszczenia. Sytuacja taka nastąpi chociażby wtedy, gdy transakcja o niższym priorytecie zarezerwuje wcześniej zasoby potrzebne do wykonania transakcji o wyższym priorytecie.

Istnieje kilka znanych propozycji rozwiązania tego problemu. Pierwszym z nich jest „strategia promocji oczekującego” (ang. *Wait Promote Strategy*) [StaZao 1988]. Według jej założeń, jeśli transakcja T_1 (o większym priorytecie) oczekuje na zasoby zarezerwowane przez transakcję T_2 (o niższym priorytecie), to transakcja T_2 musi zostać wypromowana do priorytetu transakcji T_1 , przez co szybciej zakończy swoje działanie zwalniając wymagane zasoby. Wadą tej strategii jest to, iż w dalszym ciągu przyzwala na to, aby transakcja o niższym priorytecie blokowała wykonanie istotniejszych zadań.

W przeciwieństwie do niej „strategia wysokiego priorytetu” (ang. *High Priority Strategy*) [AbbGar 1992] przerywa działanie transakcji T_2 , dzięki czemu transakcja T_1 może dokończyć swoje zadanie. Rozwiązuje to problem „odwróconych priorytetów” jednakże wprowadza dodatkowy – restartowana transakcja T_2 , ze względu na opóźnienia w realizacji, może uzyskać teraz priorytet wyższy niż transakcja T_1 , co w efekcie może zaowocować cykliczną serią wywłaszczeń transakcji. Jest to wysoce niepożądana sytuacja. W związku z tym powstało oczywiste rozszerzenie tej strategii nazwane „strategią wysokiego priorytetu bez cykli”. Rozszerzenie to wprowadza dodatkowy warunek konieczny: przerwanie wykonania transakcji T_2 może zostać zrealizowane jeżeli nowo nadany priorytet dla T_2 będzie dalej niższy niż priorytet transakcji T_1 [AbbGar 1992].

Kolejną, bardziej liberalną strategią jest „strategia warunkowego restartu” (ang. *Conditional Restart Strategy*). Jej dodatkowym atutem jest to, iż sprawdza, czy bufor czasowy (określający ilość czasu, podczas którego dana transakcja nie będzie wykonywana, a mimo wszystko zdąży wykonać się jeszcze na czas) transakcji T_1 jest większy niż estymowany czas potrzebny na ukończenie blokującej transakcji T_2 . Jest to o tyle ryzykowna strategia, że wykonanie transakcji T_2 również może być obarczone konfliktami z innymi transakcjami przez co szacowanie czasu jej wykonania jest jeszcze trudniejsze.

Problem zakleszczenia w SZBDCR również może się pojawić, zwłaszcza wtedy, gdy nie jest używany protokół zarządzania transakcjami zabezpieczający przed nim. Protokół taki jest wysoce niepożądany w SZBDCR ze względu na to, iż jest bardzo kosztowny w zasobach i czasie działania. Należy więc zastanowić się nad odpowiednią techniką wybierania tak zwanej „transakcji – ofiary”, która zostanie przerwana w celu rozwiązania konfliktu.

W publikacji [HuStTo 1990] zaproponowano następujące podejście: „Przerwij działanie transakcji, która już przekroczyła czas swojego działania. Jeśli takiej nie ma wybierz tę, która ma jeszcze najwięcej czasu na realizację swoich zadań. Alternatywnie można wybrać transakcję o mniejszym priorytecie lub krytyczności.”

4. PODSUMOWANIE

W pracy przedstawiono koncepcję SZBDCR, scharakteryzowano problemy wynikające z wprowadzenia ograniczeń czasowych. SZBDCR są to systemy zaprojektowane do pracy przy dużym obciążeniu z danymi, których stan zmienia się nieustająco. Umożliwiają one w pełni kontrolowanie czasu przetwarzania transakcji. Dzięki temu korzyści wynikające z zastosowania takiego rozwiązania są na tyle duże, iż aktualnie prowadzone są intensywne badania zarówno teoretyczne, jak i praktyczne na pierwszych, prototypowych rozwiązaniach badawczych tego rodzaju systemów. Bardzo istotnym jest precyzyjny dobór algorytmu szeregującego transakcje czasu rzeczywistego, który odbiega znacząco od tradycyjnych algorytmów szeregujących. Jednakże z uwagi na mnogość parametrów mających wpływ zarówno na jego działanie, jak i konieczność empirycznego weryfikowania własności zastosowanego rozwiązania, jest to aktualnie bardzo żywy temat, nad którym pracują wspólnie środowiska naukowe i biznesowe. Zastosowanie tych systemów może się okazać przełomowe w wielu dziedzinach.

LITERATURA

- [AbbGar 1992] Abbott R., Garcia-Molina H., *Scheduling Real-Time Transactions: a Performance Evaluation*, Nowy Jork, NY, USA, ACM, 1992.
- [Buchma 2005] Buchmann A., *Real Time Database Systems. Encyclopedia of Database Technologies and Applications*, Idea Group, 2005.
- [CapJoh 1999] Capron H.L., Johnson J.A., *Computers: Tools for an Information Age*, Boston, MA, USA, Addison-Wesley, 1999.
- [GarKao 1995] Garcia-Molina H., Kao B., *An Overview of Real-Time Database Systems*. Nato Advanced Study Institute on Real-Time Computing, 1995.
- [HuStTo 1990] Huang J., Stankovic J., Towsley D., Ramamritham K., *Real-Time Transaction Processing: Design, Implementation and Performance Evaluation*. University of Massachusetts COINS TR, 1990.
- [PetSil 1985] Peterson J.L., Silberschatz A., *Operating System Concepts*. Boston, MA, USA, Addison-Wesley, 1985.
- [ShRaLe 1988] Sha L., Rajkumar R., Lehoczky J.P., *Concurrency Control for Distributed Real-Time Databases*. ACM SIGMOD Record, 1988.
- [StaZao 1988] Stankovic J., Zhao W., *On Real-time transactions*. ACM SIGMOD Record, 1988.

REAL TIME DATABASE MANAGEMENT SYSTEM

Real Time Database Management Systems (RTDBMS) are an alternative to traditional – relational Database Management Systems (DBMS). They are designed and built for the purpose of increasing effectiveness of transactions processing, to control executing time and life cycle of real time transactions. The transaction's processing time in RTBDMS is known what makes output results useable for next proceeding in real time based systems. This document presents problems and characteristic of difficulties connected with time-restricted transaction processing. There are presented as well examples of scheduling algorithms for real time transactions.

*baza danych,
gra komputerowa cRPG,
Microsoft Access*

Hanna MAZUR*
Zbigniew GÓRAWSKI

GRY KOMPUTEROWE W PROGRAMIE MICROSOFT ACCESS – PROJEKT I IMPLEMENTACJA

W pracy opisano nietypowe wykorzystanie systemu zarządzania bazą danych Microsoft Access do utworzenia systemu bazy danych dla gry komputerowej typu cRPG. Opracowany system umożliwia tworzenie plansz, postaci, zadań i dialogów oraz wzajemnych powiązań między nimi. Dane przechowywane w bazie danych są wyświetlane na ekranie w atrakcyjnej, graficznej formie poprzez formularze. Projekt prezentuje nieszablony i innowacyjny sposób użycia środowiska MS Access.

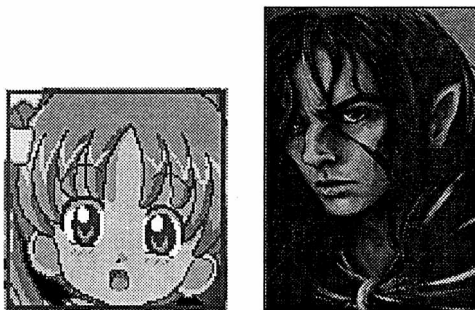
1. WPROWADZENIE

Microsoft Access to system do zarządzania relacyjnymi bazami danych wchodzący w skład pakietu biurowego Microsoft Office [MSDN 2008]. Rozbudowana funkcjonalność tego programu pozwala jednak tworzyć projekty niezwiązane ściśle z pracą biurową. Przykładem takiego niecodziennego użycia mogą być systemy *Access Gate* (autor: Zbigniew Górawski) i *Saga2006* (autorzy: Paweł Górzyński i Łukasz Skierski) – fabularne gry komputerowe wykonane w opisywanym środowisku bazodanowym w ramach zajęć projektowych z systemów baz danych.

Obie gry (*Access Gate* i *Saga2006*) są grami typu cRPG (ang. *computer Role Playing Game*), w której gracz, kierując bohaterem, wykonuje różnorodne zadania poznając świat gry i odkrywając tajniki fabuły. W grach cRPG każdy awatar (uczestnik świata wirtualnego czyli zarówno gracz jak i postać z gry) ma przypisane liczbowe współczynniki, modyfikowane w trakcie gry, które w dużej mierze zastępują umiejętności samego gracza w determinowaniu wyniku podejmowanych akcji.

* Instytut Informatyki, Wydział Informatyki i Zarządzania Politechniki Wrocławskiej, 50-370 Wrocław, Wybrzeże Wyspiańskiego 27, hanna.mazur@pwr.wroc.pl

Awatar przyjmuje rozmaite formy np. symboliczną (rys. 1) – wówczas przedstawiany jest za pomocą prostej grafiki rastrowej (bitmapowej), lub jako złożony model trójwymiarowy.



Rys. 1. Awatary bitmapowe

Awatarowi można przypisać imię, nazwisko lub pseudonim (nick) oraz określić jego wygląd (np. płeć, wzrost, wagę, kolor włosów i oczu, ubranie, umiejętności, cechy charakteru).

Awatar zazwyczaj podróżuje i ma do wykonania określone questy (zadania, misje). Po wykonaniu zadania otrzymuje nagrodę w postaci pieniędzy, punktów doświadczenia (PD) lub rzeczy. Questy mogą być główne lub poboczne. Questy główne mogą mieć strukturę liniową lub hierarchiczną – ich wykonanie zależy wówczas od podjętych decyzji. Dodatkowe zadania czyli questy poboczne na ogół nie mają wpływu na fabułę gry, ale pozwalają awatarowi zdobyć dodatkowe nagrody.

Współczynniki (zwane też statystykami lub charakterystykami) przypisywane postaci to na ogół wyrażone liczbowo cechy naturalne postaci (np. kolor oczu), wady i zalety, moce, które umożliwiają postaci wykonywanie określonych zadań, umiejętności wyuczone oraz wiedza. W wielu grach współczynniki te są od siebie zależne.

Istotnym elementem gier cRPG jest stworzenie świata (ang. *setting*) gry (fantastycznego, historycznego, baśniowego, kosmicznego itp.) przez Mistrza gry. Awatary podróżują w tym świecie, pokonują przeszkody (nawet takie jak zamknięte drzwi), używają różne rodzaje broni, przeobrażają się itp. W grze wykorzystywane są mapy i labirynty. Mistrz Gry wplątuje awatary w różne przygody i uwalnia je z nich.

Ruchome obrazy opracowuje się wykorzystując animację lub technikę przechwytywania ruchów (ang. *motion capture*), w której na postacię przenosi się nagrane (dzięki nałożonym czujnikom na nogi, ręce, głowę) ruchy ludzi.

Wydawałoby się, że te dwa byty – fabularna gra komputerowa cRPG i system zarządzania bazą danych MS Access nie mają ze sobą nic wspólnego. Jednak fascynacja grami fabularnymi oraz konieczność wykonania systemu bazodanowego zmusiła

i zachęciła autorów systemów do podjęcia próby w kierunku połączenia tych dwóch zgoła odmiennych światów.

Wstępna analiza dowolnej gry typu cRPG wykazuje, że w grach tych przechowuje się bardzo dużo danych, które na początku są przydzielane np. losowo a następnie aktualizowane. Dobrze zaprojektowana baza danych ułatwia wizualizację powiązań między danymi i planowanie rozwoju gry.

Wybrany do tego celu (ze względu na dostępność, prostotę i łatwość implementacji) system MS Access umożliwia zapisanie wymaganych cech o postaciach i świecie gry w tabelach bazy danych (takich jak *Osoby* czy *Zadania*), elementów graficznych – w plikach graficznych lub jako obiekty OLE. Obie aplikacje *Access Gate* i *Saga2006* wykonano w oparciu o oferowane możliwości formularzy programu MS Access [Access 2008, Forum 2008].

2. PROJEKT BAZY DANYCH

Zaprojektowanie dedykowanej bazy danych przechowującej dane potrzebne do gry nie było wcale proste głównie ze względu na nietypowość i specyfikę zadania (np. słownictwo). Przy pierwszym spojrzeniu na planowaną grę wydawało się, że MS Access nie jest odpowiednim narzędziem do jej implementacji a sama gra nie ma wielkiego związku z bazami danych. Dopiero po głębszej analizie okazało się, że dane wykorzystywane w grze można przechowywać w bazie danych i MS Access można wykorzystać do implementacji prototypu gry.

Celem projektu było usprawnienie sesji gry fabularnej opartej na systemie AD&D (ang. *Advanced Dungeons and Dragons*) poprzez opracowanie wirtualnej karty gracza, zawierającej wszystkie statystyki. Podczas sesji, każdy gracz kieruje swoją postacią (zwaną BG czyli bohaterem gracza), którą charakteryzuje sześć podstawowych atrybutów: siła, zręczność i budowa (zaliczane do atrybutów fizycznych) oraz intelekt, roztropność i charyzma (czyli atrybuty mentalne).

W bazie danych musi być tabela przechowująca dane dotyczące osób znajdujących się w grze (tzw. NPC – *Non Playable Character*) czyli postaci zaludniających świat gry, ale których role odgrywa komputer, oraz informacje o miejscu przebywania każdej osoby, jej imię, portret oraz krótką biografię. Kolejne dane przechowywane w bazie to dane o zadaniach – sytuacjach w grze, których rozwiązywanie odsłania przed graczem fabułę gry. Z powyższymi danymi wiążą się dialogi, czyli kwestie wypowiedziane przez konkretne osoby w zadanej sytuacji (na danym etapie fabularnym zadania). Niezbędna jest również tabela przechowująca dane dotyczące poszczególnych plansz – ich grafikę, nazwę czy miejsce, w którym pojawia się na niej główny bohater oraz miejsca na mapie, do których gracz nie ma dostępu (np. rejon głębokiej wody czy wysokich gór), a także tabela do przecho-

wywnia informacji dotyczących miejsc na mapie, które umożliwiają podróże pomiędzy planszami.

Ostatecznie zaprojektowane bazy danych umożliwiają dodawanie nowych awatarów (zarówno graczy jak i postaci fabularnych np. potwora, czarodzieja, wojownika, księżniczki itd.), zmianę atrybutów postaci, śledzenie rozwoju postaci, sprzedaż i kupno nowych rzeczy, przeglądanie danych, sporządzanie raportów (np. na temat pokonanych wrogów), sporządzanie statystyk dla gracza i dla Mistrza Gry.

Obie bazy danych mogą być rozbudowywane o dodatkowe elementy, takie jak dźwięki i muzyka (np. mówione kwestie postaci). Ponieważ celem projektu nie była pełna implementacja gry a jedynie wykonanie prototypu w celu sprawdzenia możliwości nietypowego wykorzystania programu MS Access, zatem ograniczono się do realizacji wyżej opisanej, podstawowej funkcjonalności systemu umożliwiającej między innymi wykonywanie na obiektach bazy operacji CRUD (Create, Read, Update, Delete) oraz realizacji głównych zadań gry.

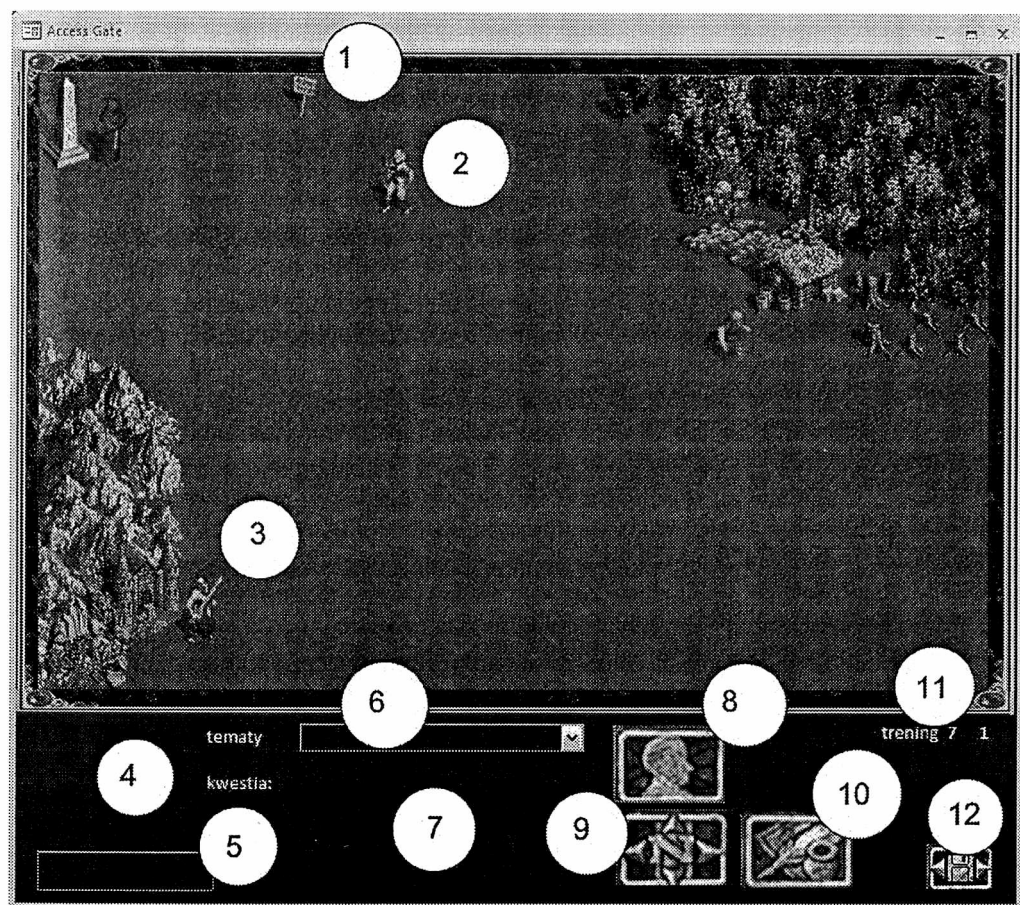
3. IMPLEMENTACJA PROJEKTU

Całą implementację projektu gier wykonano w programie Microsoft Access 2007. W przykładowej grze *Access Gate* opartej na stworzonym silniku wykorzystano dodatkowo grafikę z gry *Heroes of Might and Magic III* [Herosi 2008] oraz *Baldur's Gate* [Baldur 2008]. Główne okna gier wykonano przy użyciu formularzy. Zastosowano standardowe elementy pakietu takie jak etykiety, pola tekstowe, listy rozwijane i inne. Mechanizm obsługi zdarzeń został napisany w języku Visual Basic, z wyjątkiem najprostszych komend, utworzonych przy pomocy mechanizmu makr.

Tabela 1. Legenda dla głównego okna gry *Access Gate*

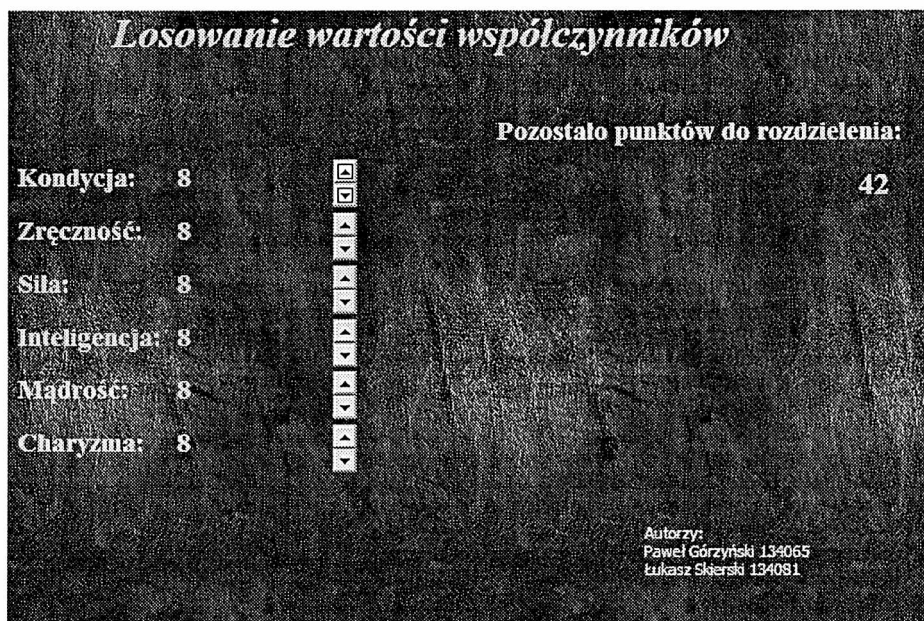
Lp.	Legenda
1	Miejsce przejścia do innej lokalizacji
2	Główny bohater
3	Postać z gry (NPC)
4	Miejsce na portret postaci z gry
5	Imię awatara
6	Menu wyboru tematów rozmowy
7	Kwestie wypowiadane przez postacie
8	Przycisk rozmowy
9	Przycisk podróży
10	Przycisk dziennika zadań
11	Nazwa aktualnej mapy oraz bieżąca pozycja bohatera
12	Powrót do menu głównego

Główne okno gry *Access Gate*, wraz z opisem znajdujących się tam elementów, jest przedstawione na rysunku 2. W tabeli 1 przedstawiono opis zaznaczonych elementów z rysunku 2 przedstawiającego przykładową planszę gry *Access Gate*.

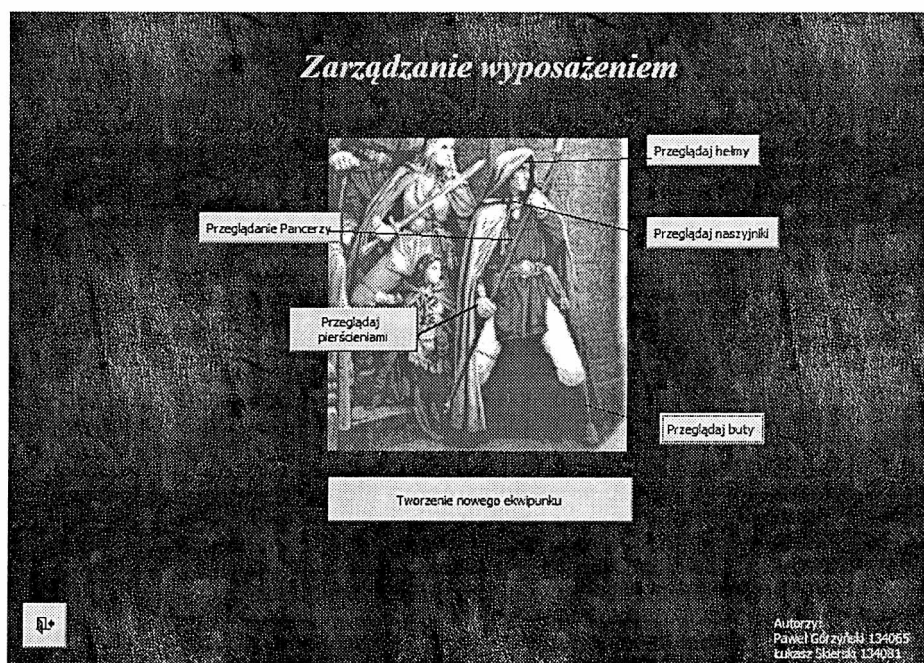


Rys. 2. Główne okno gry *Access Gate*

Kolejne formularze, łączące w sobie zarówno elementy graficzne jak i tekstowe służą do wykonywania określonych funkcji systemu. Ich przykłady przedstawiono na rysunkach 3 i 4. Celem artykułu nie jest ich pełne przedstawienie a jedynie zasygnalizowanie możliwości takiego podejścia. Dzięki kreatorowi formularzy w sposób szybki i wygodny uzyskano czytelny wygląd oraz automatyczną implementację podstawowych funkcjonalności takich jak dodawanie, edycja czy usuwanie danych.

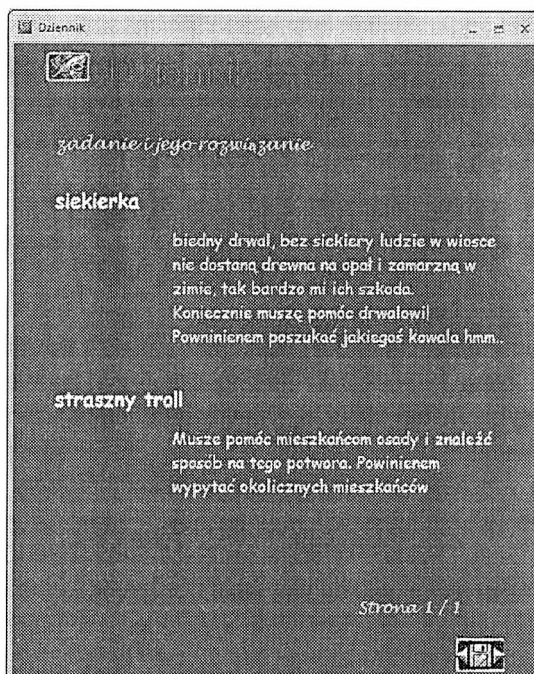


Rys. 3. Przykładowy ekran systemu *Saga 2006* do przeglądania współczynników



Rys. 4. Formularz systemu *Saga 2006* do nadawania wartości współczynników

Elementem składowym gry są również tematyczne zestawienia, grupujące i przedstawiające w czytelnej formie wybrane informacje dotyczące świata gry. Na rysunku 5 przedstawiono przykładową listę – dziennik, zawierający spis aktualnie wykonywanych zadań wraz z krótkim komentarzem. Do wykonania tego typu zestawień (przedstawianych w formie tekstowej i wykresów) użyto narzędzi do tworzenia raportów. Podobnie jak wyżej opisane formularze, każdy z raportów jest uruchamiany jako osobne okno. Dzięki takiemu rozwiązaniu udało się lepiej dopasować elementy graficzne oraz przybliżyć wygląd aplikacji do ekranów gier.



Rys. 5. Przykładowy raport – dziennik zadań z gry *Access Gate*

Pewnego omówienia wymaga typ danych przechowujących dane graficzne. Naturalnym wyborem jest pole typu *OLE Object*, które służy do przechowywania obiektów takich jak plik, obraz, dźwięk, wykres o rozmiarze do 2 GB (przy czym przechowywanie tak dużych danych nie jest zalecane gdyż powoduje bardzo wolne działanie bazy). Pola tego typu tworzą obrazy map bitowych plików oryginalnych a następnie wyświetlają te mapy w tabelach, formularzach i raportach. W tym celu na komputerze, na którym jest uruchamiana baza danych musi być zarejestrowany serwer OLE, obsługujący pliki tego typu. Przy niektórych konfiguracjach sprzętowo-programowych przechowywane obrazy są wyświetlane niepoprawnie bądź wcale (problem dotyczy np. plików JPEG). Ponadto przechowywanie w bazie plików graficznych bardzo zwiększa jej roz-

miar. Z uwagi na powyższe trudności wykorzystane pola załącznikowe. Pliki graficzne przechowywane są poza bazą danych, a w tabeli przechowywane są jedynie nazwy odpowiednich plików lub ścieżki dostępu do plików. Sposób ten zapewnia bardziej wydajną pracę i nie jest zależny od zarejestrowanych serwerów OLE.

Po opracowaniu systemów-gier autorzy przeprowadzili mały test. Osobom, które nic nie wiedziały na temat gier cRPG, przedstawiły to zagadnienie. Niestety, ze względu na bardzo specjalizowane słownictwo, słuchacze nie rozumieli o co w tym wszystkim chodzi. Dopiero przedstawienie systemu (a właściwie jego prototypu) pomogło zrozumieć ideę przedstawianych gier komputerowych. Opracowana wizualizacja sprawiła, że omawiane pojęcia stały się czytelne i zrozumiałe dla osób, które dotychczas nie znały gier fabularnych ani obowiązujących w nich zasad. Zaimplementowane systemy sprawiły radość również ich autorom, którzy osiągnęli swój cel, zaprojektowali relacyjną bazę danych dla fabularnych gier komputerowych i wykonali prototypy tych gier w systemie MS Access.

4. PODSUMOWANIE

Opracowane systemy bazodanowe *Access Gate* i *Saga2006* dla gier fabularnych są przykładem nieszablonowego podejścia do zagadnień związanych z projektowaniem i implementacją relacyjnych baz danych. Obie implementacje pokazały, że wykorzystując standardowe mechanizmy i techniki popularnego programu jakim jest system zarządzania bazą danych MS Access, uzyskano niestandardowy rezultat – fabularną grę komputerową. Do tworzenia gier komputerowych są opracowane specjalizowane narzędzia np. OpenGL czy Direct3D (jako część biblioteki DirectX). Dla producentów gier komputerowych niezwykle ważne jest stałe obniżanie kosztów i czasu produkcji. Przygotowanie komercyjnego projektu w MS Access jest nierealne ze względu na ograniczenia techniczne i wydajnościowe, ale przedstawione przykłady pokazują, że system Access może być wykorzystany na przykład do szybkiego prototypowania wczesnych wersji fabularnych gier komputerowych lub do zbudowania bazy danych dla gry.

LITERATURA

- [Access 2008] <http://www.access.vis.pl/>
- [Baldur 2008] [http://pl.wikipedia.org/wiki/Baldur's_Gate_\(seria\)](http://pl.wikipedia.org/wiki/Baldur's_Gate_(seria))
- [Forum 2008] <http://www.access-programmers.co.uk/forums/>
- [Herosi 2008] http://pl.wikipedia.org/wiki/Heroes_of_Might_and_Magic_III
- [MSDN 2008] <http://msdn.microsoft.com>
- [Wiki 2008] [http://pl.wikipedia.org/wiki/Awatar_\(rzeczywisto%C5%9B%C4%87_wirtualna\)](http://pl.wikipedia.org/wiki/Awatar_(rzeczywisto%C5%9B%C4%87_wirtualna))

COMPUTER GAMES IN MICROSOFT ACCESS – DESIGN AND IMPLEMENTATION

This paper describes a non-typical utilisation of the Microsoft Access database management system: for creation of a system for computer Role Playing Game (cRPG). The system allows creating charts, characters, tasks, dialogues and relationships between them. Data stored in the database is displayed on the screen in an attractive, graphic way, based on forms. The project presents an original and innovative manner of using the MS Access environment.

*mobilne bazy danych,
mechanizmy synchronizacji danych,
Microsoft SQL Server 2005 Mobile,
Sybase SQL Anywhere 10, Oracle Lite 10g*

Rafał BOBORYCKI
Artur WILCZEK*

CHARAKTERYSTYKA SYSTEMÓW ZARZĄDZANIA BAZAMI DANYCH DLA URZĄDZEŃ PRZENOŚNYCH

W artykule scharakteryzowano mobilne bazy danych oraz omówiono istotne problemy związane z ich wykorzystaniem w urządzeniach przenośnych. Tytułem wprowadzenia w tematykę mobilnych baz danych, przedstawiono ogólne zagadnienia i podano podstawową terminologię stosowaną w tej dziedzinie. Omówiono problemy współbieżnego dostępu do danych i dostępne sposoby wymiany danych pomiędzy bazami mobilnymi. Zaprezentowano trzy wybrane systemy zarządzania bazami danych dla urządzeń przenośnych: *Microsoft SQL Server 2005 Mobile*, *Oracle Lite 10g* i *Sybase SQL Anywhere 10*. Opisano architekturę ich mechanizmów synchronizacji danych, narzędzia wspomagające obsługę infrastruktury mobilnej oraz metody detekcji konfliktów podczas synchronizacji danych. Oceniono także rozwiązania producentów w świetle dopasowania do potrzeb przenośnych urządzeń, na których zostaną zastosowane.

1. WPROWADZENIE

Postęp technologiczny w dziedzinie elektroniki i informatyki, jaki dokonał się w ostatniej dekadzie zaowocował pojawieniem się nowych odmian komputerów. Po komputerach przenośnych takich jak laptopy i notebooki pojawiły się nowe urządzenia przenośne, które dzięki miniaturyzacji, mieszczą się z łatwością w dłoni lub kieszeni. Takie urządzenia dysponują często mocą obliczeniową, jaką niedawno dysponowały stacjonarne komputery osobiste, co umożliwia obsługiwanie znacznie rozbudowanych aplikacji. Niektóre modele wyposażone są w moduł telefonu komórkowego, a także system nawigacji satelitarnej GPS.

* Instytut Informatyki, Wydział Informatyki i Zarządzania Politechniki Wrocławskiej, 50-370 Wrocław, Wybrzeże Wyspiańskiego 27, artur.wilczek@pwr.wroc.pl

To sprawiło, że zalety i potencjał komputerów przenośnych szybko zostały dostrzeżone w nowych zastosowaniach. Wykorzystanie takich urządzeń pozwala na dostęp do danych poza siedzibą organizacji, a więc na przykład: w trakcie podróży służbowych. Takie zastosowanie urządzeń mobilnych staje się popularne na przykład w firmach kurierskich oraz firmach opartych o infrastrukturę terenowych przedstawicieli handlowych.

Aby sprostać nowym potrzebom rynku producenci oprogramowania musieli zaofiarować nowe narzędzia, w tym umożliwić bezpieczne i sprawne zarządzanie danymi znajdującymi się na tych urządzeniach. Czołowi producenci systemów baz danych, tacy jak *Oracle*, *Microsoft*, *IBM* czy *Sybase* dostarczyli narzędzia niezbędne do wspierania celów organizacji wykorzystujących komputery przenośne w swojej działalności.

2. CHARAKTERYSTYKA MOBILNYCH BAZ DANYCH

Charakterystykę mobilnych baz danych rozpoczniemy od przedstawienia ich zastosowań.

2.1. ZASTOSOWANIA MOBILNYCH BAZ DANYCH

Mobilne bazy danych znalazły zastosowanie tam, gdzie przy użyciu komputerów przenośnych potrzebny jest zdalny dostęp do danych organizacji, a gdy dostęp ten jest z różnych przyczyn niemożliwy.

W wielu przypadkach dostęp ten zapewnia stała łączność bezprzewodowa (Wi-Fi, GPRS, etc.). Jednak często zdarza się, że urządzenie znajduje się poza zasięgiem takiej sieci, a wtedy wymiana danych z centralnym serwerem jest niemożliwa. Istotnym ograniczeniem dzisiejszych sieci bezprzewodowych jest także szybkość transferu danych. Może okazać się, że działanie aplikacji, która aby wykonać jakąkolwiek operację musi połączyć się za pośrednictwem sieci bezprzewodowej z serwerem w celu pobrania lub wysłania danych, jest zbyt powolne. Istnieją również zastosowania komputerów przenośnych, w których dane są poufne. Użycie połączenia bezprzewodowego do przesyłania takich danych mogłoby doprowadzić do ich przechwycenia. Za użyciem mobilnej bazy danych zainstalowanej na urządzeniu przenośnym zamiast wykorzystania klasycznej architektury klient-serwer przemawia również fakt, że w niektórych przypadkach aplikacje na ogół pracują na statycznych danych (np. dane adresowe klienta zmieniają się dość rzadko). W tym wypadku bardziej racjonalne wydaje się być składowanie takich danych na urządzeniu, niż każdorazowe przesyłanie ich przez sieć.

Podsumowując, zastosowanie mobilnych baz danych na urządzeniach przenośnych daje użytkownikom możliwość pełnego przetwarzania danych – ma on wrażenie jak by był stałe połączony z serwerem swojej organizacji, będąc nawet poza zasięgiem wszelkich sieci bezprzewodowych.

2.2. RODZAJE SYNCHRONIZACJI DANYCH

W mobilnych bazach danych stosowane są różne metody synchronizacji danych z serwerem centralnym. Wyróżnić można kilka rodzajów synchronizacji danych względu na [Bębel 2003], [Zyskowski 2004]:

- tryb synchronizacji,
- zakres synchronizacji,
- kierunek zainicjowania.

Ze względu na tryb, synchronizację dzielimy na synchroniczną i asynchroniczną. W pierwszym przypadku synchronizacja z Wydawcą następuje natychmiast po modyfikacji danych Subskrybenta. W drugim przypadku, Subskrybent jest synchronizowany na żądanie użytkownika (tzw. odświeżanie manualne), albo automatycznie przez system zarządzania bazą danych – okresowo, z zadanyim interwałem czasowym.

Ze względu na zakres, synchronizację dzielimy na pełną i przyrostową. Synchronizacja pełna polega na skopiowaniu całej zawartości Publikacji do Subskrybenta. Natomiast przy synchronizacji przyrostowej do Subskrybenta kopiowane są wyłącznie zmiany dokonane w Publikacji od czasu ostatniej synchronizacji. Zmiany te nazywane są deltami.

Ze względu na kierunek zainicjowania, synchronizację dzielimy na „push” – aktywną stroną synchronizacji danych jest Wydawca oraz „pull” – aktywną stroną synchronizacji danych jest Subskrybent.

2.3. DETEKCJA I ROZWIĄZYWANIE KONFLIKTÓW SYNCHRONIZACJI DANYCH

Po zainicjowaniu synchronizacji, zmiany dokonane w jednej bazie danych (np. Subskrybenta) mogą kolidować z danymi, które zostały zmienione w na innym Subskrybencie lub Wydawcy. Stosowane są dwie metody detekcji błędów:

- detekcja na poziomie wierszy tabeli,
- detekcja na poziomie kolumn tabeli.

Aby zobrazować zagadnienie, przedstawimy prosty przykład. Przypuśćmy, że baza danych A jest bazą Wydawcy, natomiast baza B to baza Subskrybenta. Obie bazy

dysponują początkowo takim samym zbiorem danych. W zbiorze tym znajduje się tabela T, której wycinek zawartości widoczny jest na rysunku 1.

Na bazie danych A, dokonana zostaje operacja uaktualnienia danych, w wyniku której zmodyfikowana zostaje wartość kolumny *LastName*, w wierszu pierwszym tabeli T (*Achong*). Z kolei na bazie danych B, w tym samym wierszu dokonana zostaje zmiana w kolumnie *EmailAddress* (*gustavo0@adventure-works.com*). Zmiany przedstawiono na rysunku 2.

ID	Title	FirstName	MiddleName	LastName	EmailAddress
1	Mr.	Gustavo	NULL	Achong	gustavo0@adventure-works.com
2	Ms.	Catherine	R.	Abel	catherine0@adventure-works.com
3	Ms.	Kim	NULL	Abercrombie	kim2@adventure-works.com
4	Sr.	Humberto	NULL	Acevedo	humberto0@adventure-works.com
5	Sra.	Pilar	NULL	Ackerman	pilar1@adventure-works.com

Rys. 1. Tabela T przed dokonaniem modyfikacji

ID	Title	FirstName	MiddleName	LastName	EmailAddress
1	Mr.	Gustavo	NULL	<u>Boborycki</u>	<u>rafal.boborycki@student.pwr.wroc.pl</u>
2	Ms.	Catherine	R.	Abel	catherine0@adventure-works.com
3	Ms.	Kim	NULL	Abercrombie	kim2@adventure-works.com
4	Sr.	Humberto	NULL	Acevedo	humberto0@adventure-works.com
5	Sra.	Pilar	NULL	Ackerman	pilar1@adventure-works.com

Rys. 2. Tabela T po dokonaniu zmian. Podkreślenie linią przerywaną oznacza dane zmodyfikowane w bazie A, linią ciągłą – dane zmodyfikowane w bazie B

Przy detekcji na poziomie wierszy, gdy wywołana zostanie procedura synchronizacji baz A i B, wykryty zostanie konflikt, ponieważ zmiany dokonane na tabeli T nastąpiły w tym samym wierszu. Uruchomiona zostanie zatem procedura rozwiązywania konfliktów przez odpowiedni mechanizm procesu synchronizacji, która rozstrzygnie, które dane należy zaakceptować, a które usunąć.

Detekcja na poziomie kolumn, jest nowocześniejsza i bardziej wyrafinowana. Nawet jeśli zmiany dokonane zostaną w tym samym wierszu tabeli T, mechanizm jest w stanie rozpoznać miejsce dokonania tej zmiany. W omawianym przykładzie synchronizacja baz A i B nastąpi więc bez konieczności uruchamiania procedury rozwiązywania konfliktów. Metoda ta jest także trudniejsza do technicznej realizacji – nie

wszyscy dostawcy systemów zarządzania bazami danych posiadają jej implementację w swoich mechanizmach synchronizacji.

Jest wiele metod rozwiązywania konfliktów podczas synchronizacji danych. Powszechnie stosowana jest metoda najpóźniejszego znacznika (ang. *latest timestamp*), opisana w [Bębel 2003]: „Metoda najpóźniejszego znacznika bazuje na znaczniku czasowym, który dla każdej operacji aktualizacji danych określa moment jej wystąpienia. W przypadku konfliktu dwóch operacji *UPDATE* wykonana zostaje ta, której znacznik czasowy jest późniejszy. Dobrym rozwiązaniem jest połączenie metody najpóźniejszego znacznika z inną metodą, która przejmie rozwiązanie konfliktu w przypadku, gdy konfliktowe operacje mają identyczne znaczniki czasowe. Metoda wymaga zaimplementowania mechanizmu oznaczania czasu wykonania każdej operacji *UPDATE* danych replikowanej tabeli (np. odpowiedniego wyzwalacza)”. Gdy aktualizacje danych nastąpiły w tym samym czasie, wykorzystywane są następujące przykładowe metody rozwiązania konfliktu:

1. Jeżeli konflikt nastąpił między Subskrybentem i Wydawcą, stosuje się prostą zasadę, że zawsze wygrywa Subskrybent, lub zawsze wygrywa Wydawca.
2. Jeżeli konflikt pojawia się między dwoma Subskrybentami, zachowane zostają dane tego Subskrybenta, który jako pierwszy dokonał synchronizacji danych z Wydawcą.
3. Innym sposobem rozwiązania konfliktu między dwoma Subskrybentami jest nadanie im priorytetów. Zachowane zostają dane tego Subskrybenta, który miał najwyższy priorytet. Jeżeli priorytety są równe, konflikt rozstrzygany jest według zasady nr 2.

Producenci przedstawionych w niniejszej pracy systemów umożliwiają również zdefiniowanie własnych reguł rozwiązywania konfliktów synchronizacji danych. Pozwala to na optymalne dostosowanie mechanizmu synchronizacji do indywidualnych potrzeb i sytuacji, w jakich będzie wykorzystany.

2.4. OGÓLNY SCHEMAT SYNCHRONIZACJI DANYCH

Przyglądając się rozwiązaniom, jakie oferują dostawcy systemów baz danych dla urządzeń przenośnych w dziedzinie tworzenia i konfigurowania synchronizacji, można wyodrębnić pewien schemat, według którego przebiega ten proces:

- utworzenie Publikacji,
- skonfigurowanie mechanizmu synchronizacji,
- utworzenie Subskrypcji.

Tworząc Publikację, określa się zbiór obiektów bazy danych (Wydawcy) – tabele, kolumny, procedury składowane etc. – jakie mają zostać skopiowane podczas pierwszej synchronizacji (zreplikowane) do innej bazy (Subskrybent), i które mogą być zmieniane przez operacje na tej bazie. Zmiany te będą mieć odzwierciedlenie po kolejnej synchronizacji.

Podczas konfigurowania mechanizmu synchronizacji, określa się typ i rodzaj Replikacji, sposoby detekcji i rozwiązywania konfliktów, a także dokonuje się ustawień w zakresie technicznej realizacji (adresy komputerów, hasła, ustawienie katalogów itd.).

Podczas tworzenia Subskrypcji, aktywuje się na wybranej bazie (lub tworzy się nową, o specjalnych parametrach) zdolność do transferu danych z Wydawcą.

3. ARCHITEKTURA I FUNKCJE WYBRANYCH SYSTEMÓW

W tym rozdziale przedstawione i omówione zostaną narzędzia oferowane przez *Microsoft*, *Oracle* i *Sybase*, jednak z uwagi na tematykę pracy opis ograniczony zostanie do wybranych funkcjonalności związanych z zarządzaniem i obsługą baz mobilnych oraz mechanizmów synchronizacji danych. Bardziej uszczegółowiony opis znajduje się w [Boborycki 2008].

1.1. MICROSOFT SQL SERVER 2005 MOBILE

Firma Microsoft jest dostawcą systemu mobilnego o nazwie Microsoft SQL Server 2005 Mobile.

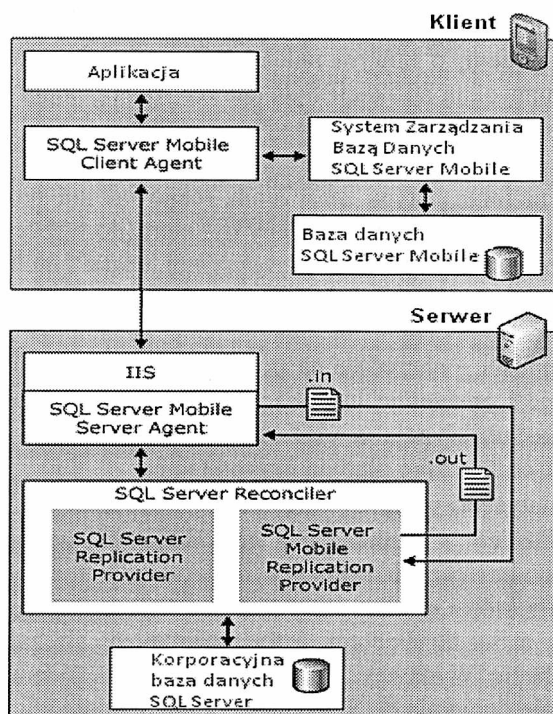
3.1.1. MECHANIZM SYNCHRONIZACJI DANYCH

Proces synchronizacji danych odbywa się za pośrednictwem następujących komponentów *Microsoft SQL Server 2005 Mobile* [Microsoft]:

- *SZBD SQL Server Mobile* – system zarządzający lokalną bazą danych na urządzeniu przenośnym. Śledzi wszelkie zmiany dokonane na bazie danych (operacje *INSERT*, *UPDATE*, *DELETE*).
- *SQL Server Mobile Client Agent* – najważniejszy komponent mechanizmu replikacji na urządzeniu przenośnym. Realizuje on specjalny interfejs, który wywoływany jest przez aplikacje w celu programowej obsługi mechanizmu replikacji.
- *SQL Server Mobile Server Agent* – odpowiada za zarządzanie procesem komunikacji między systemami zarządzania bazami danych *SQL Server* i *SQL Server Mobile*. Znajduje się na komputerze, na którym uruchomiony jest serwer *IIS*

i przechwytuje wszelkie żądania wysyłane protokołem HTTP przez *SQL Server Mobile Client*.

- *SQL Server Mobile Replication Provider* – wywoływany jest przez komponent *SQL Server Reconciler*, gdy rozpoczyna się proces synchronizacji danych. Oba komponenty znajdują się na komputerze, na którym uruchomiony jest serwer *IIS*.



Rys. 3. Schemat komponentów *Microsoft SQL Server 2005 Mobile* uczestniczących w procesie synchronizacji danych [Microsoft]

3.1.2. KOMPONENTY SYSTEMU

W zasadzie wszystkie funkcje oferowane przez *SQL Server 2005* zgrupowane są w jednej aplikacji o nazwie *SQL Server Management Studio*. *SQL Server* wyposażony został w liczne funkcje ułatwiające szeroko pojęte administrowanie mobilną bazą danych w analogiczny sposób jak bazą korporacyjną. Posiada więc system wizualizacji danych i obiektów bazy, umożliwiający ich przetwarzanie i edycję, okno interpretera

języka SQL, narzędzie optymalizacji zapytań *QueryAnalyzer*, z możliwością prezentacji planu wykonania, narzędzie śledzące zapytania do bazy z poziomu aplikacji (*SQL Server Profiler*) i wiele innych. Specjalna wersja *QueryAnalyzer*'a może zostać także zainstalowana bezpośrednio na urządzeniu przenośnym. W razie problemów mobilną bazą danych, może być bardzo pomocnym narzędziem diagnostycznym. Istnieje także możliwość przechowywania wyników pracy w postaci projektów, co ułatwia i porządkuje prace z bazami danych. Istnieją również wbudowane mechanizmy do integracji z bazami danych *Oracle*.

Kolejnym komponentem, o którym należy wspomnieć jest *Replication Monitor*, uruchamiany z poziomu okna aplikacji *SQL Server Management Studio*. Umożliwia on śledzenie parametrów Replikacji poprzez [Microsoft]:

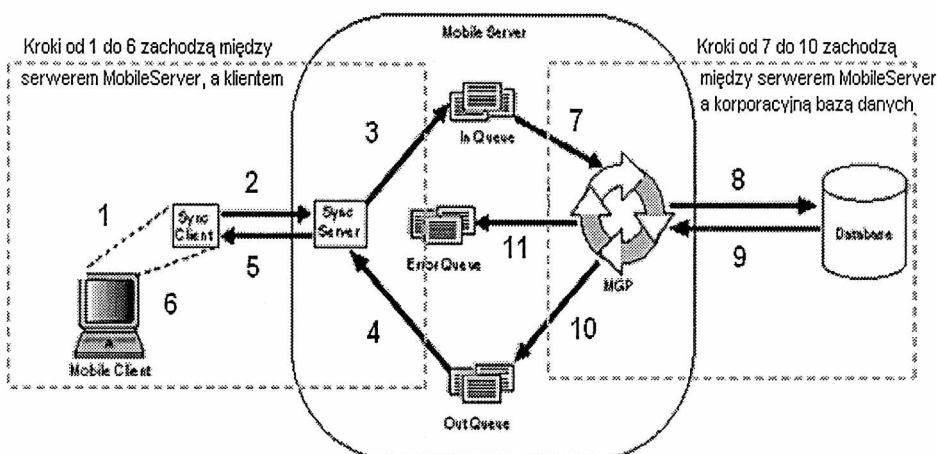
- Ustawianie powiadomień o zagrożeniach i ich wartości progowych – możliwe jest wybranie, na jakiego typu zagrożenia, reagować ma system i jaka ich wartość ma być osiągnięta, aby generowany był alert (do wyboru m.in. przekroczenie czasu opóźnienia w replikacji transakcyjnej, czasu synchronizacji, nieudana próba przetworzenia zadanej ilości wierszy w określonym czasie w replikacji skalującej).
- Pomiar wydajności – narzędzie na podstawie gromadzonych danych automatycznie generuje i prezentuje informacje na temat wydajności osiągananej przez pary Publikacja-Subskrypcja w pięciostopniowej skali (*Excellent, Good, Fair, Poor, Critical*), co znacznie ułatwia interpretację.
- Szczegółowe statystyki przeprowadzanych synchronizacji.
- Określanie opóźnień w replikacjach transakcyjnych za pomocą znaczników – do danych podlegających replikacji dodawana jest niewielka ilość danych (ang. *token*), dla której dokonywany jest pomiar czasu przebycia drogi pomiędzy Wydawcą, a Subskrybentem. Informacje te zostają zapisane w dzienniku transakcji na bazie Wydawcy. Pozwala to również zidentyfikować sytuację, w której dane nie docierają do Subskrybenta lub Wydawcy.
- Możliwość podglądu przebiegu transakcji i pomiar czasu jej trwania w replikacjach transakcyjnych.

3.2. ORACLE LITE 10G R2

Firma Oracle jest dostawcą systemu o nazwie Oracle Lite 10G R2.

3.2.1. MECHANIZM SYNCHRONIZACJI DANYCH

Schemat przedstawiony na rysunku 4 ilustruje wykorzystanie komponentów i przebieg procesu synchronizacji danych w *Oracle Database Lite* [Oracle 1], [Oracle2].



Rys. 4. Schemat komponentów *Oracle Database Lite* uczestniczących w procesie synchronizacji danych [Oracle1], [Oracle2]

Procesu synchronizacji danych przebiega w następujących krokach.

1. Synchronizacja danych inicjowana jest przez użytkownika komputera przenośnego (mobilnego klienta) lub przez mechanizm automatycznej synchronizacji danych.
2. Na mobilnym kliencie gromadzone są informacje o wszelkich modyfikacjach danych w lokalnej bazie. Informacje te wysyłane są za pośrednictwem komponentów *Sync Client* i *Sync Server* do głównego serwera synchronizacji danych (*Mobile Server*).
3. *Sync Server* organizuje odebrane informacje w kolejkę (*In-Queue*).
4. *Sync Server* odbiera i gromadzi informacje o modyfikacjach danych, jakie mają zostać przeprowadzone na mobilnym kliencie z kolejki *Out-Queue*.
5. *Sync Client* pobiera modyfikacje w postaci pliku *ODB*.
6. Mobilny klient wprowadza zmiany do bazy danych *Oracle Lite* (baza zostaje automatycznie utworzona, jeśli synchronizacja odbywa się po raz pierwszy).
7. Wszystkie modyfikacje danych przeprowadzone przez mobilnych klientów kolekcjonuje *Message Generator and Preprocessor* (MGP).
8. MGP wprowadza modyfikacje do odpowiednich tabel korporacyjnej bazy danych.
9. MGP przygotowuje dane, które mają zostać przekazane mobilnym klientom.
10. Dane zostają umieszczone w kolejce (*Out-Queue*) i oczekują na synchronizację, podczas której *Sync Server* przekaże je do mobilnego klienta.

11. Wszelkie błędy, jakie pojawiły się podczas wykonywania kroków od 7 do 10 gromadzone są w postaci kolejki *ErrorQueue*.

3.2.2. KOMPONENTY SYSTEMU

W odróżnieniu od *SQL Server Management Studio*, firma *Oracle* zdecydowała się na zastosowanie kilku mniejszych aplikacji spełniających określone funkcje. W skład *Lite 10g* wchodzi serwer dla baz mobilnych i replikacji danych *Mobile Server*, środowisko do tworzenia projektów opartych o mobilne bazy danych *Mobile Database Workbench* oraz udogodnienie, którego nie posiadają produkty pozostałych producentów – system wspomagający instalowanie aplikacji wraz z mobilną bazą danych na urządzeniu przenośnym za pośrednictwem Internetu – *Packaging Wizard* wraz z *Mobile Client Setup*. Zarządzanie obsługiwanymi aktualnie mobilnymi klientami wspomaga aplikacja *Mobile Manager*, której interfejs także jest dostępny przez Internet.

Aby użyć *Oracle Mobile Manager* należy wpisać w przeglądarkę adres http://nazwa_komputera:8888/webtogo/index.html i zalogować się na konto administratora baz danych. Aplikacja oferuje między innymi następujące funkcjonalności:

- alerty o zagrożeniach lub błędach (m.in. o synchronizacji), na które należy zwrócić uwagę, istnieje także możliwość przeglądania i redagowania dziennika błędów;
- administrowanie mobilnymi bazami danych, z którymi prowadzona jest wymiana informacji. Możliwe jest na przykład, poprzez klikanie w odpowiednie linki, zsynchronizowanie danej, mobilnej bazy danych z serwerem centralnym (o ile w danej chwili możliwe jest połączenie internetowe);
- administrowanie urządzeniami, na których mobilne bazy fizycznie się znajdują: np. zainstalowanie nowej wersji oprogramowania na danym urządzeniu, bądź pobranie aktualnych danych o jego stanie (dostępna ilość miejsca na dyskach, zainstalowane bazy danych etc.);
- administrowanie uprawnieniami użytkowników, którzy mają dostęp do mobilnej bazy danych, w tym zmiana zbioru danych na samej bazie, do której użytkownik ma dostęp przez aplikację.

MSQL to aplikacja dostarczająca prostego interfejsu (linia komand) do obsługi mobilnej wersji systemu zarządzania bazą danych. Instalowana jest wraz innymi komponentami na kliencie podczas pierwszej synchronizacji z *Oracle Mobile Server*. W razie awarii na urządzeniu, może pomóc w znalezieniu jej przyczyny.

Mobile Database Workbench jest w produkcie *Oracle Lite 10g* odpowiednikiem *Microsoft SQL Server Management Studio*, jeśli chodzi o funkcjonalność obsługi mobilnych baz danych. Także tutaj możliwe jest zakładanie projektów organizujących

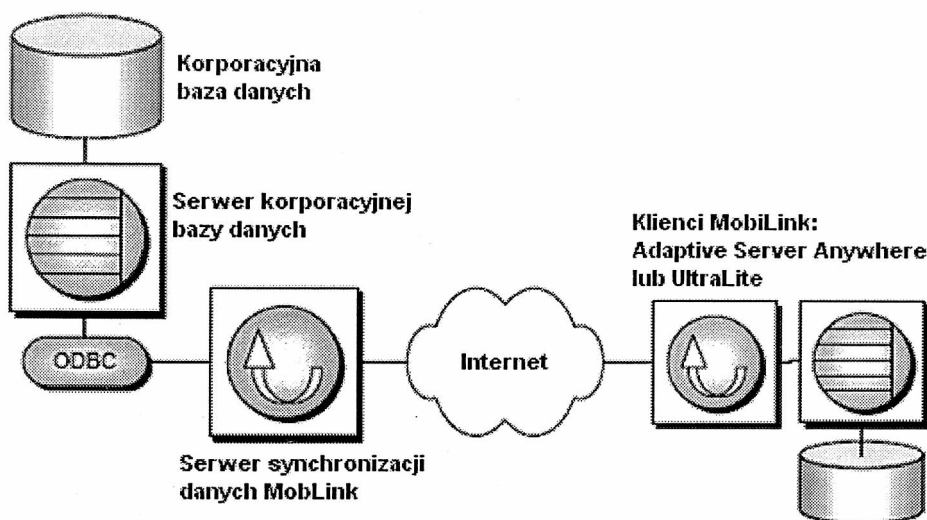
skrypty SQL, tworzenie publikacji przy użyciu wyspecjalizowanych narzędzi oraz administrowanie infrastrukturą replikacji.

3.3. SYBASE SQL ANYWHERE 10

Firma Sybase jest dostawcą systemu o nazwie *Sybase SQL Anywhere 10*.

3.3.1. MECHANIZM SYNCHRONIZACJI DANYCH

Na rysunku 5 zamieszczono diagram, który przedstawia najważniejsze komponenty systemu synchronizacji [Sybase].



Rys. 5. Schemat komponentów *Sybase SQL Anywhere* uczestniczących w procesie synchronizacji danych [Sybase]

Na diagramie zamieszczono następujące elementy:

- korporacyjna baza danych – stanowi centralną kopię wszystkich danych, jakie biorą udział w procesie synchronizacji;
- serwer korporacyjny – serwer lub system zarządzania bazą danych, który zarządza korporacyjną bazą danych. W roli korporacyjnej bazy danych może występować produkt *Sybase*, np. *Adaptive Server Anywhere* lub system innego producenta (*Oracle*, *Microsoft*, *IBM DB2*);

- połączenie *ODBC* – za jego pośrednictwem realizowane są wszystkie połączenia pomiędzy serwerem *MobiLink*, a korporacyjną bazą danych;
- serwer *MobiLink* – zarządza procesem synchronizacji danych i stanowi interfejs, za pomocą którego mobilni klienci komunikują się z serwerem korporacyjnym.
- klient *MobiLink* – może być instalowany na komputerze kieszonkowym, telefonie, serwerze lub komputerze klasy PC.

3.3.2. KOMPONENTY SYSTEMU

Podobnie jak w systemie *Microsoft* w *SQL Management Studio*, większość funkcjonalności systemu *Sybase SQL Anywhere 10* zgrupowana jest w jednej aplikacji o nazwie *Sybase Central*. Funkcjonalność również jest podobna – zarządzanie bazami danych, obiektami w bazie, użytkownikami, prawami dostępu itp. Nie istnieje jednak możliwość zakładania projektów, a rolę interpretera poleceń *SQL* pełni oddzielna aplikacja – *Interactive SQL*, którą można uruchomić z poziomu *Sybase Central*. Funkcjonalnością, którą nie posiadają produkty konkurentów, jest wygodna i szybka technika kopiowania obiektów bazy danych metodą przeciągnij i upuść.

4. PODSUMOWANIE

W ramach pracy przedstawiono trzy systemy zarządzania bazami danych dla urządzeń przenośnych: *Microsoft SQL Server 2005 Mobile*, *Oracle Lite 10g* i *Sybase SQL Anywhere 10*.

Próbę oceny narzędzi przeznaczonych do konfigurowania mechanizmu synchronizacji znaleźć można w pracy [Boborycki 2008]. Najbardziej przyjaznym i najłatwiejszym w obsłudze i poznaniu wydaje się być system *Oracle Lite 10g*. Używając *Oracle Packaging Wizard*, użytkownik nie jest absorbowany koniecznością konfiguracji stricte technicznej strony wymiany danych jak np. podawanie adresów internetowych i numerów portów, definiowanie uprawnień do katalogów, konfigurowanie ustawień serwera wymiany danych itp. Wszystkie te czynności wykonywane są automatycznie. W zasadzie konieczna wiedza do obsługi kreatora ogranicza się do znajomości podstawowych danych logowania (login i hasło) koniecznych do komunikacji z bazą i z serwerem. Użytkownik definiuje Publikację wskazując tabele bazy danych, i zaznacza typ platformy docelowej, na którą ma być dostarczona mobilna baza danych. Ma do dyspozycji także opcję, której nie posiadają rozwiązania pozostałych producentów – wraz z bazą na urządzenie docelowe może być także dostarczana aplikacja, która będzie z niej korzystać.

Kreator synchronizacji systemu *Sybase SQL Anywhere* nie jest już tak intuicyjny, ale na podstawie bardziej wnikliwej analizy dokumentacji można również w akcepto-

walnym czasie zainicjować pierwszą wymianę danych między Wydawcą, a Subskrybentem. Rozczarowuje natomiast *Microsoft SQL Server Management Studio*. Wykorzystanie aż trzech kreatorów, mocno angażujących użytkownika w konfigurację elementów technicznych architektury sprawia, że informacje zawarte w dokumentacji są niewystarczające i dodatkowych wskazówek trzeba szukać na internetowych listach dyskusyjnych.

Z narzędzi dodatkowych, wspomagających obsługę mobilnej infrastruktury wymienić należy *Oracle Mobile Manager*, pozwalające na instalację mobilnej bazy danych oraz aplikacji z Internetu i ułatwiające ich administrowaniem w okresie eksploatacji. *Microsoft SQL Management Studio* pozwala natomiast na połączenie się z bazą danych znajdującą się fizycznie na urządzeniu przenośnym, co również usprawnia diagnostykę i ułatwia działania administracyjne.

Przy ocenie mechanizmów synchronizacji nie były brane pod uwagę takie aspekty jak architektura ich komponentów. Bardziej racjonalne wydaje się być porównanie tego, co oferują w kwestii praktycznego tworzenia i obsługi rozwiązań mobilnych.

Należy więc tutaj przyrzeć się metodom detekcji konfliktów podczas synchronizacji: produkty firm *Sybase* i *Microsoft* oferują nowocześniejsze i bardziej precyzyjny mechanizm – detekcji na poziomie kolumn (omówienie sposobów detekcji znajduje się w rozdziale 2.3), podczas gdy *Oracle* zapewnia detekcję tylko na poziomie wierszy. Oznacza to, że podczas synchronizowania danych przy użyciu mechanizmu *Oracle* częściej wywoływane będą metody ich rozwiązywania – niektóre zmienione dane będą nadpisywane, występuje więc grubsza ziarnistość aktualizowania danych i związana z tym niższa skuteczność.

Ocenić można także rozwiązania producentów w świetle dopasowania do potrzeb przenośnych urządzeń, na których zostaną zastosowane. *Oracle* i *Microsoft* niezależnie czy jest to komputer kieszonkowy o niskiej wydajności i niewielkich zasobach pamięci, czy też przenośny komputer *PC* jak notebook wielokrotnie przewyższający go osiągnięciami, oferują tylko jeden rodzaj systemu zarządzania bazą danych. Maksymalny rozmiar danych w obu przypadkach to 4 GB. *Sybase* dzięki innemu podejściu do instancji Wydawcy i Subskrybenta, zapewnia znacznie większą możliwość przystosowania: system *UltraLite* dedykowany jest komputerom przenośnym i odznacza się niskim zapotrzebowaniem na zasoby systemowe oferując maksymalnie 2 GB rozmiaru przechowywanych danych. Gdy konieczna jest większa pojemność, np. w przypadku notebooków można zastosować system *iAnywhere* bądź *SQL Anywhere*.

Jest jeszcze jeden aspekt, w którym rozwiązania poszczególnych producentów się różnią: obsługa systemów zarządzania bazami danych innych producentów. Aby utworzyć infrastrukturę wymiany danych, w przypadku *Oracle* bazy danych Wydawcy i Subskrybenta muszą być rozwiązaniami tej właśnie firmy (np. *10g* i *Lite 10g*). Nie dopuszcza się sytuacji, w której serwer korporacyjny z bazą Wydawcy obsługiwany byłby przez innego dostawcę bazy danych, a infrastrukturę wymiany danych i systemy zarządzania dla baz mobilnych dostarczane były przez *Oracle*. Taką możliwość ofe-

ruje *Microsoft*: można synchronizować urządzenia przenośne z systemami *SQL Server Mobile* z korporacyjną bazą *Oracle*. Najwięcej możliwości pod tym względem oferuje firma *Sybase*: dla rozpatrywanego przypadku rolę systemu zarządzania bazy danych na serwerze korporacyjnym może pełnić produkt nie tylko firmy *Oracle* ale również *Microsoft*, a także *IBM (DB2)*.

LITERATURA

- [Bębel 2003] Bębel B., Wrembel R., *Replikacja danych w bazach danych Oracle 9i*, 2003
http://www.ploug.org.pl/seminarium/seminarium_VII/pliki/5a.pdf
- [Boborycki 2008] Boborycki R., *Systemy zarządzania bazami danych dla urządzeń przenośnych*, magisterska praca dyplomowa, Politechnika Wrocławska 2008
- [Microsoft] Dokumentacja techniczna Microsoft, *SQL Server 2005 Books Online (September 2007)*
<http://msdn2.microsoft.com/en-us/library/ms152768.aspx>
- [Oracle1] Dokumentacja techniczna Oracle, *Adding Mobile Capability to an Enterprise Application With Oracle Database Lite*, 2007
<http://www.oraclewhitepapers.com/>
- [Oracle2] Dokumentacja techniczna Oracle, *Oracle Database Lite: Synchronizing Data Between a Device and an Oracle Database*, 2007
<http://www.oraclewhitepapers.com/>
- [Sybase] Dokumentacja techniczna Sybase, *Overview of UltraLite Development and Mobi-Link Synchronization*, 2005
<http://www.sybase.com/detail?id=1038632>
- [Zyskowski 2004] Zyskowski R., *Replikacja SQL Server*, materiał z prezentacji podczas Microsoft Professional Developer Days, 2004
<http://www.microsoft.com/poland/developer/pdd2004/default.mspx>

OVERVIEW OF COMMERCIAL DATABASE MANAGEMENT SYSTEMS FOR MOBILE DEVICES

This article presents a brief overview of mobile databases with the key aspects of their usage in mobile devices. As an introduction into mobile databases, some basic terminology and facts are presented. Problems of concurrent data access and available communication methods between databases are described. Three database management systems for mobile devices are covered: Microsoft SQL Server 2005 Mobile, Oracle Lite 10g and Sybase SQL Anywhere 10. We describe architectures of these databases, data synchronization mechanisms, maintenance tools for mobile infrastructure and conflicts detection methods during data synchronization. Finally, we evaluate the aforementioned solutions with respect to the needs for mobile devices.

*mobilne bazy danych,
testy baz danych,
Microsoft SQL Server 2005 Mobile,
Sybase SQL Anywhere 10,
Oracle Lite 10g*

Rafał BOBORYCKI
Artur WILCZEK*

PORÓWNANIE WYDAJNOŚCI WYBRANYCH KOMERCYJNYCH SYSTEMÓW ZARZĄDZANIA BAZAMI DANYCH DLA URZĄDZEŃ PRZENOŚNYCH

W artykule prezentowane są wyniki testów wydajności trzech systemów zarządzania bazami danych: *Oracle Lite 10g*, *Microsoft SQL Server 2005 Mobile*, *Sybase SQL Anywhere 10*. Na potrzeby badań przygotowana została testowa aplikacja korzystająca z mobilnej bazy danych. Aplikację tę zaprojektowano na wzór komercyjnych systemów dla firm działających w oparciu o sieć przedstawicieli handlowych, gdzie mobilne bazy danych mają częste zastosowanie. Celem jej wykonania było poznanie i pokazanie najbardziej przydatnych oraz najczęściej wykorzystywanych mechanizmów badanych produktów. Opracowano różne wersje aplikacji testowej, które powstały w celu przeprowadzenia badań w różnych środowiskach – na komputerach kieszonkowych z systemem *Windows Mobile 5.0*, oraz na przenośnych komputerach osobistych z systemem *Windows XP*. W oparciu o zaproponowane jednolite kryteria przeprowadzone zostały testy porównawcze trzech badanych systemów. Na podstawie analizy uzyskanych wyników wskazane zostały słabe oraz silne strony poszczególnych systemów.

1. WPROWADZENIE

Mobilne bazy danych dostosowane do działania na urządzeniach przenośnych zachowują pełną funkcjonalność systemu zarządzania bazami danych. Rozwiązania mobilne oferowane są przez wszystkich największych dostawców systemów zarządzania bazami danych: *Oracle*, *Microsoft*, *IBM* (produkt *DB2 Everyplace 9.1*), *Sybase*. Pojawiają się również nowe produkty jak *Birdstep RDM Server 8.0*. Do celów pracy wy-

* Instytut Informatyki, Wydział Informatyki i Zarządzania Politechniki Wrocławskiej, 50-370 Wrocław, Wyrbrzeże Wyspiańskiego 27, artur.wilczek@pwr.wroc.pl

brane zostały trzy spośród wyżej wymienionych: *Microsoft SQL Server 2005 Mobile*, *Oracle Lite 10g R2* oraz *Sybase SQL Anywhere 10*.

Microsoft, *Oracle* i *Sybase* oferują kolejną już generację swoich produktów. Premiery testowanych wersji miały miejsce kilka lat przed powstaniem niniejszego artykułu (rok 2008), zostały więc dobrze poznane przez programistów, administratorów i specjalistów od systemów bazodanowych. Dostępna jest obszerna dokumentacja oraz duża ilość materiałów pomocniczych ułatwiających pracę z badanymi środowiskami. Wiadomo także, że trwają prace nad kolejnymi generacjami rozwiązań u wszystkich trzech producentów.

Producenci systemów zarządzania bazami danych chcąc zaoferować swój produkt na urządzenia przenośne zdawali sobie sprawę, że korporacjom nie wystarczy sama możliwość składowania danych na urządzeniach przenośnych. Oprócz tego należało zapewnić mechanizmy pozwalające na transmisje danych pomiędzy urządzeniem mobilnym, a systemem centralnym. Pojawia się tutaj problem zapewnienia spójności i aktualności danych. Jeżeli bowiem na różnych urządzeniach istnieje kopia tych samych danych i w tym samym czasie zostaną one zmienione w różny sposób, to należy rozstrzygnąć, które dane mają wrócić na serwer centralny i być uważane jako aktualne.

Dodatkowym utrudnieniem jest – mimo wszystkich ich zalet – bardzo niska wydajność komputerów przenośnych (w szczególności kieszonkowych) w porównaniu z maszynami stacjonarnymi, wyposażonymi niejednokrotnie w kilka procesorów, dużą ilość pamięci operacyjnej i wiele dysków twardych.

Wszystkie te kwestie sprawiły, że stworzenie dobrego oprogramowania umożliwiającego pracę mobilnym bazom danych stało się trudnym wyzwaniem.

2. BADANIE WYDAJNOŚCI WYBRANYCH SYSTEMÓW

W celu poznania właściwości badanych systemów od strony praktycznej, wykonane zostały trzy aplikacje testujące. Pierwsza zaprojektowana i wykonana została na wzór istniejących na rynku, komercyjnych systemów dla firm działających w oparciu o sieć przedstawicieli handlowych, gdzie mobilne bazy danych mają najczęstsze zastosowanie (dokładny opis aplikacji w [Boborycki 2008]). Doświadczenia z prac nad aplikacją pozwoliły wyodrębnić te funkcje i mechanizmy systemów zarządzania mobilnymi bazami danych, które są najczęściej wykorzystywane w praktyce przez firmy dostarczające oprogramowanie na urządzenia przenośne. Pod kątem badania tych własności, opracowana została metoda testów wydajnościowych oparta o jednolite kryteria porównawcze. Realizacja testów była możliwa dzięki przygotowaniu dwóch aplikacji – jednej działającej na komputerach kieszonkowych z systemem Windows Mobile 5 i drugiej pracującej na przenośnych komputerach osobistych z systemem Windows XP. Zastosowanie

dwóch aplikacji pozwala na zbadanie zachowania systemów w różnych środowiskach. Na podstawie analizy uzyskanych danych wskazane zostały słabe oraz silne strony poszczególnych systemów.

2.1. METODA BADAŃ

Na podstawie doświadczeń zebranych podczas prac nad przykładową aplikacją oraz stosowanych metod testowania baz danych, opracowana została metoda badań wydajnościowych, łącząca testy baz danych i testy aplikacji. Zastosowanie samych metod testowania wydajności baz danych, z pominięciem mechanizmów synchronizacji danych, nie pozwoliłoby na pełne porównanie badanych systemów. Wykorzystane zostały zatem dodatkowo wybrane metody badania wydajności aplikacji bazy danych, aplikacji webowych oraz metody testów нефункциональных. Scenariusz testów został tak opracowany, aby najbardziej odpowiadał on charakterystyce pracy aplikacji współdziałającej z mobilną bazą danych w różnych środowiskach. Wyodrębniony został podział na cztery rodzaje testów (na podstawie [Meier 2007], [Rauterberg 1992], [Splain 2001], [Dustin 2002], [Hnatkowska 2006], [WWW1]:

- testy przeciążenia (ang. *stress testing*),
- testy obciążeniowe (ang. *load testing*),
- testy wydajnościowe (ang. *performance testing*),
- testy pojemności (ang. *volume testing*).

W artykule zaprezentowano wyniki wybranych testów wydajnościowych. Pozostałe wyniki z wyżej wymienionych grup testów znaleźć można w pracy [Boborycki 2008].

2.2. WYNIKI TESTÓW

W niniejszym rozdziale przedstawimy opisy i wyniki przeprowadzonych testów wydajnościowych: synchronizacji danych, czasu wykonania zapytań oraz czasu wykonania modyfikacji danych.

2.2.1. TEST SYNCHRONIZACJI DANYCH

Cel testu:

Ocena wydajności mechanizmów synchronizacji poszczególnych systemów zarządzania bazami danych.

Opis:

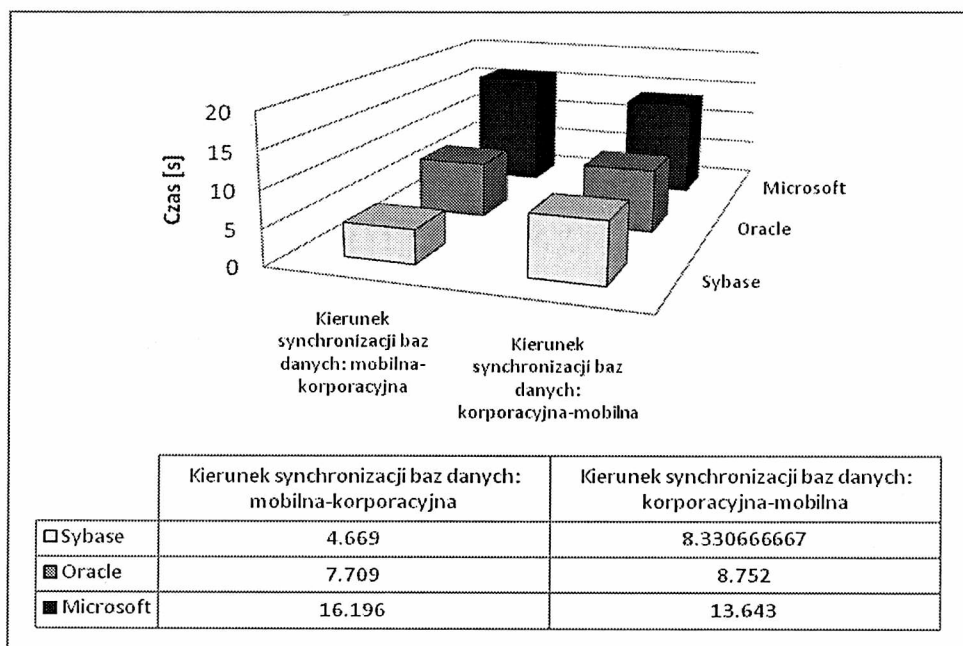
Synchronizacja danych inicjowana była przez aplikację testującą z komputera przenośnego, na którym znajdowała się mobilna baza danych (Subskrybent). W roli

komputera przenośnego występował komputer kieszonkowy lub komputer PC. Komputer kieszonkowy (system operacyjny *Windows Mobile 5.0*) połączony był serwerem Wydawcy za pośrednictwem kabla *ActiveSync*, natomiast komputer PC (system operacyjny *Windows XP*) siecią LAN.

Przed rozpoczęciem pomiarów czasu, w ramach procedury testowej, dokonywana była dwukrotna synchronizacja danych, w celu zapewnienia identycznego zbioru danych na Subskrybencie i Wydawcy. Pomiędzy krokami 3, 4, 5 i 6 prezentowanej niżej procedury testowej zachowane były odstępy czasowe, wynoszące 2 minuty, w celu zapewnienia prawidłowego uaktualnienia Publikacji przez mechanizmy synchronizacji.

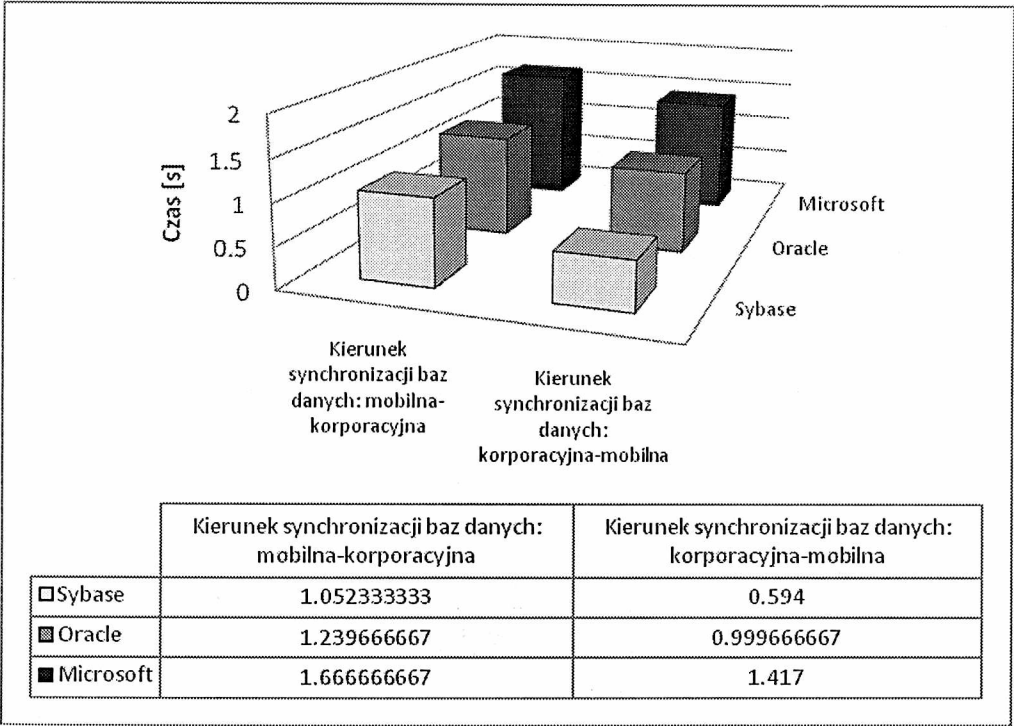
Modyfikacja danych na Subskrybencie dokonywana była przez uruchomienie wcześniej przygotowanej procedury. Wynikiem jej działania była zmiana danych w trzech kolumnach tabeli dla 443 wierszy (pkt. 3 procedury testowej). Zakres zmian dobrany był w taki sposób, że całkowity rozmiar synchronizowanych danych był zachowany (zmiany na poziomie pojedynczych znaków w zmiennych typu *string* i cyfr w zmiennych typu *int*). Po dokonaniu synchronizacji Subskrybenta z Wydawcą (pkt. 4 procedury testowej), następowało odtworzenie pierwotnego wyglądu danych sprzed ich zmodyfikowaniem przygotowaną procedurą (pkt. 5 procedury testowej). Po dokonaniu kolejnej synchronizacji, dane miały wartości takie jak przed rozpoczęciem testu.

Wyniki testów przedstawiono na rysunkach 1 i 2.



Rys. 1. Wykres czasu synchronizacji danych między mobilnym Subskrybentem (komputer kieszonkowy z systemem *Windows Mobile 5.0*), a Wydawcą (baza korporacyjna)

- 1. Uruchomienie aplikacji testującej na Subskrybencie i programowe zainicjowanie synchronizacji danych, zarejestrowanie czasu trwania synchronizacji (Subskrybent → Wydawca), weryfikacja uaktualnienia danych na Wydawcy.
- 2. Modyfikacja danych na Wydawcy.
- 3. Uruchomienie aplikacji testującej na Subskrybencie i programowe zainicjowanie synchronizacji danych, zarejestrowanie czasu trwania synchronizacji (Wydawca → Subskrybent), weryfikacja uaktualnienia danych na Subskrybencie.
- 4. Dwukrotne powtórzenie kroków 3–6.



Rys. 2. Wykres czasu synchronizacji danych między mobilnym Subskrybentem (komputer kieszonkowy z systemem *Windows XP*), a Wydawcą (baza korporacyjna)

Interpretacja wyników:

Najkrótsze czasy synchronizacji danych w obu kierunkach na badanej platformie testowej uzyskano dla produktu *Sybase SQL Anywhere*. Uzyskane wyniki nie pozwalają stwierdzić, czy czas synchronizacji danych zależy od kierunku, w którym się odbywa (przesłanki do hipotezy o takiej zależności dostarczają wyniki uzyskane dla produktów *Microsoft*).

2.2.2. TEST CZASU WYKONANIA ZAPYTAŃ

Cel testu:

Porównanie czasu wykonania zapytań tym samym zbiorze danych, przez różne systemy zarządzania bazami danych.

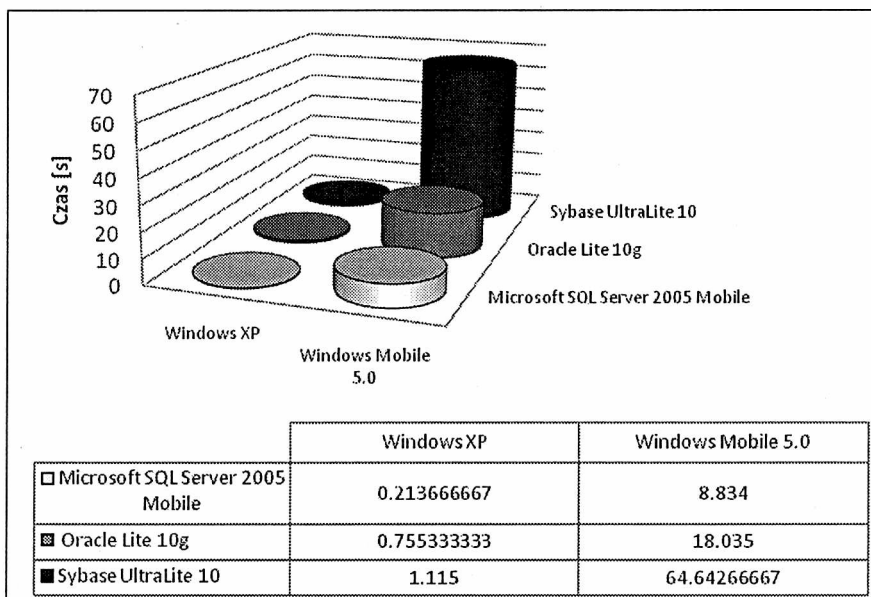
Opis:

Pomiar czasu wykonania zapytania nie uwzględniał otwierania i zamykania połączenia z bazą danych. Zapytanie testowe zostało tak skonstruowane, aby wykorzystywane były funkcje konwertujące, operatory porównywania, w tym wyszukiwania wzorca. Zwracane są 443 wiersze.

Procedura testowa:

1. Skopiowanie szablonu mobilnej bazy danych (uruchomienie serwera baz danych), uruchomienie aplikacji testującej.
2. Wykonanie zapytania testowego.
3. Zarejestrowanie czasu.
4. Dwukrotne powtórzenie kroków 2–3.
5. Zatrzymanie aplikacji, usunięcie mobilnej bazy danych (zatrzymanie serwera baz danych).

Wyniki testów przedstawiono na rysunku 3.



Rys. 3. Wykres obrazujący czas wykonania zapytania testowego

Interpretacja wyników:

Wyniki tego testu potwierdzają zależność osiąganych czasów wykonania zapytań od mocy obliczeniowej środowiska testowego. Daje się także zauważyć wyraźna przewaga produktów *Microsoft* i *Oracle* nad *Sybase*.

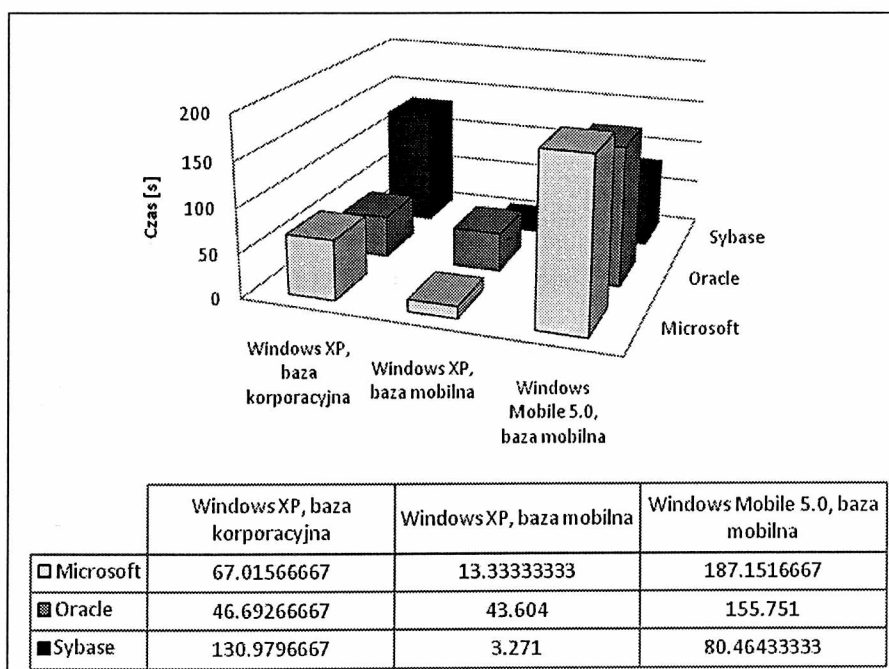
2.2.3. TEST CZASU WYKONANIA MODYFIKACJI DANYCH

Cel testu:

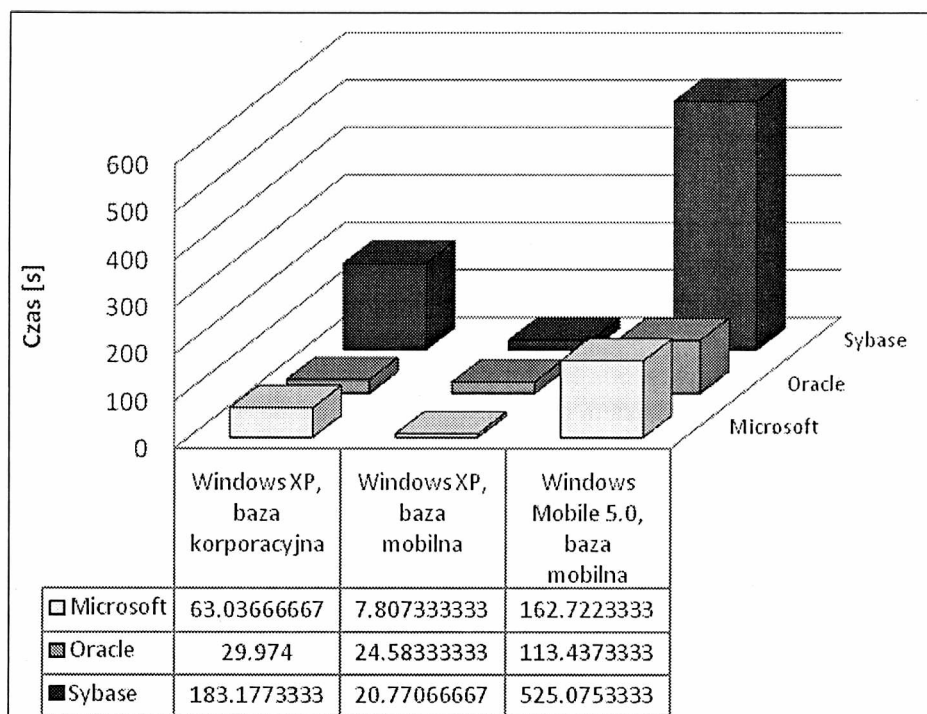
Porównanie czasu wykonania modyfikacji na tym samym zbiorze danych, przez różne systemy zarządzania bazami danych.

Opis:

Pomiar czasu wykonania zapytania nie uwzględniał otwierania i zamykania połączenia z bazą danych. Polecenie testowe `DeleteCommand.sql` usuwa, a `UpdateCommand.sql` w prosty sposób aktualizuje 1000000 wierszy w wybranej tabeli. Z uwagi na techniczne problemy synchronizacji bazy *Oracle Lite 10g* z tabelą testową o 1 000 000 wierszy na komputerze kieszonkowym, testy przeprowadzono w celach porównawczych na tabeli z 100 000 wierszy.



Rys. 4. Wykres czasu wykonania polecenia `DeleteCommand.sql`



Rys. 5. Wykres czasu wykonania polecenia UpdateCommand.sql

Procedura testowa:

1. Skopiowanie szablonu mobilnej bazy danych (uruchomienie serwera baz danych), uruchomienie aplikacji testującej.
2. Wykonanie polecenia modyfikacji danych.
3. Zarejestrowanie czasu, usunięcie mobilnej bazy (usunięcie tabeli z bazy korporacyjnej i ponowne jej utworzenie).
4. Dwukrotne powtórzenie kroków 1–3.

Wyniki testów przedstawiono na rysunkach 4 i 5.

Interpretacja wyników:

Polecenie DeleteCommand.sql najkrócej było realizowane w bazach Sybase. Sytuacja była jednak odmienna, gdy na tej samej tabeli dane były modyfikowane poleceniem UpdateCommand.sql. Baza Oracle wypadła najgorzej podczas testu na komputerze kieszonkowym. Pomimo 10-krotnie mniejszej liczby wierszy w tabeli uzyskane czasy były podobne do wyników konkurentów.

3. PODSUMOWANIE

Niżej zestawiono mocne i słabe stron badanych systemów zarządzania bazami danych dla urządzeń przenośnych:

- *Sybase SQL Anywhere 10*
 - długotrwałe przetwarzanie zapytań,
 - wydajny mechanizm synchronizacji danych,
- *Microsoft SQL Server 2005 Mobile*
 - wysoka wydajność w przetwarzaniu zapytań,
- *Oracle Lite 10g*
 - wysoka wydajność w przetwarzaniu zapytań,
 - zawodność mechanizmu synchronizacji danych,
- tendencje ogólne
 - możliwości mobilnych systemów zarządzania bazami danych w dużej mierze zależą od wydajności platformy sprzętowej, na której są zainstalowane.

Jeżeli chodzi o obsługę dużych ilości danych, najlepsze wyniki osiągnął w testach produkt *Sybase*. Przewaga nad produktami *Microsoft* oraz *Oracle* była szczególnie widoczna podczas testów synchronizacji danych. Transfer zarówno dużych rozmiarowo obiektów, jak i dużych ilości małych obiektów odbywał się najsprawniej. Najwyższą była także efektywność przyjmowania danych z aplikacji, gdy brane są pod uwagę najczęstsze zastosowania (np. wczytanie i zapisanie do bazy danych przez aplikację obszernego dokumentu). Należy jednak dodać, że maksymalny rozmiar pliku bazy danych dla *Sybase UltraLite 10* to 2 GB, podczas gdy *Oracle Lite 10g* oraz *Microsoft SQL Server 2005 Mobile* przechowują objętościowo dwa razy więcej danych. Przeprowadzenie testu zapytań na dużych tabelach, pozwoliło także wyznaczyć granicę ich rozmiaru (ilości wierszy), do której często stosowane w zapytaniach operacje łączenia tabel są wykonywane w racjonalnym czasie (do około 1s). Granica ta dla baz mobilnych wynosi – w przypadku urządzeń wykorzystanych do testów – ok. 100 000 wierszy w tabeli na notebooku z systemem *Windows XP* oraz do 10000 wierszy w tabeli na komputerze kieszonkowym z systemem *Windows Mobile 5.0*. Przekroczenie tych wartości o rząd wielkości spowoduje znaczne wydłużenie się czasu przetwarzania zapytania. Rzadziej stosowane operacje jednoczesnego uaktualniania dużych ilości danych, szybciej przebiegają na bazach mobilnych niż korporacyjnych (na tej samej platformie sprzętowej), jednak wycofanie dokonanych zmian jest niemożliwe. Podane powyżej granice są szczególnie aktualne dla produktu *Oracle*. Zapytania na bazach *Lite 10g* trwały najdłużej, a mechanizm synchronizacji zawodził zarówno po stronie mobilnego Subskrybenta (duże objętościowo dane), jak i korporacyjnej (zbyt duże wymagania sprzętowe korporacyjnego serwera baz danych w stosunku do użytego komputera).

Testy wydajnościowe przeprowadzone na przykładowych danych dla firmy z infrastrukturą rozproszoną sieci handlowej wykazały wielokrotnie szybsze przetwarzanie zapytań na bazach korporacyjnych niż na ich odpowiednikach mobilnych. Na wydajność zasadniczy wpływ ma tutaj moc obliczeniowa urządzenia, na którym pracuje aplikacja. Przetwarzanie złożonych zapytań, wykorzystujących podzapytania, funkcje

agregujące i konwertujące, operatory porównania itp. sprawiało trudności systemowi zarządzania bazą danych *Sybase UltraLite 10*. W porównaniu z pozostałymi uczestnikami testu osiągnięte czasy były znacznie wyższe, zdarzały także się przypadki, że polecenia nie udało się zrealizować z powodu zbyt długiego czasu oczekiwania.

Próby obciążeniowe systemu zarządzania mobilną bazą danych poprzez wykonywanie polecenia SQL w trakcie przetwarzania przez system innego, bardziej złożonego polecenia, wykazały najwyższą podatność *Sybase UltraLite 10*. Takie zachowanie nie było widoczne w przypadku *SQL Server 2005 Mobile* – czas wykonania polecenia testowego był zawsze podobny.

LITERATURA

- [Boborycki 2008] Boborycki R., *Systemy zarządzania bazami danych dla urządzeń przenośnych*. Praca magisterska, Politechnika Wrocławska, Wrocław 2008.
- [Dustin 2002] Dustin E., *Effective Software Testing*. Addison Wesley, 2002, ISBN 0-20179-429-2.
- [Hnatkowska 2006] Hnatkowska B., materiały do wykładu z kursu Projektowanie Systemów Informatycznych, Politechnika Wrocławska, 2006.
- [Meier 2007] Meier J.D., Fare C., Bansode P., Barber S., Rea D., *Performance Testing Guidance for Web Applications. Patterns & practices*. Microsoft Corporation, 2007.
<http://www.codeplex.com/PerfTestingGuide/Release/ProjectReleases.aspx?ReleaseId=6690>.
- [Rauterberg 1992] Rauterberg M., *Designing and testing the usability of a relational database system with end-user oriented benchmark tasks*. Elsevier Science Publisher B.V., 1992.
- [Splain 2001] Splain S., Jaskiel S.P., *The Web Testing Handbook*. STQE Publishing, 2001
<http://bdonline.sqe.com/>.
- [WWW1] Portal testerzy.org, *Rodzaje testów wydajnościowych*
<http://testerzy.org/automatyzacja/rodzaje-testow-wydajnosciowych/>.

COMMERCIAL DATABASE MANAGEMENT SYSTEMS FOR MOBILE DEVICES EFFECTIVENESS TESTS

This article contains description and comparison of effectiveness offered by *Oracle Lite 10g*, *Microsoft SQL Server 2005 Mobile*, and *Sybase SQL Anywhere 10*. Test application making use of a mobile database, as well as two testing applications were made. The testing application was designed to reflect commercial systems for companies basing on a net of sales representatives, where mobile databases are the most frequently used. The aim of its creation was recognizing and demonstrating the most useful and frequently used mechanisms of the products researched in the thesis. The two remaining applications were created to test those mechanisms in various environments – on personal digital assistants, making use of the Windows Mobile 5.0 operating system, and on portable personal computers operating on Windows XP system. With the use of common criteria a number of comparison tests on the three researched systems were carried out. Basing on the analysis of the results, strong and weak sides of tested systems were presented.

Hanna MAZUR*

Zygmunt MAZUR*

ZARZĄDZANIE WIEDZĄ A POLITYKA BEZPIECZEŃSTWA

Rozwój technologii informatycznych, informatyzacja przedsiębiorstw, gromadzenie i przesyłanie danych w postaci elektronicznej wymusza dodatkowe działania związane z ochroną informacji oraz z zorganizowanym sposobem dokumentowania i zarządzania wiedzą. Efekty udostępnienia niepowołanego dostępu do danych, utrata danych, niewłaściwy przepływ informacji czy chwilowy brak dostępu do danych może mieć poważne konsekwencje dla działalności przedsiębiorstwa. Wszystkie te elementy mogą (a nawet powinny) być regulowane przez systemy zarządzania wiedzą i przyjętą politykę bezpieczeństwa. W pracy przedstawiono podstawowe zagadnienia związane z zarządzaniem wiedzą w ramach przyjętej i realizowanej przez przedsiębiorstwo polityki bezpieczeństwa oraz systemu zarządzania bezpieczeństwem informacji (SZBI).

1. WPROWADZENIE

Gospodarka oparta na wiedzy, w której niezwykle cennym dobrem są dane (przechowywane i przesyłane za pomocą systemów teleinformatycznych), informacja i wiedza, zmusza kierownictwo firm do opracowywania i przestrzegania zasad dotyczących ochrony tych dóbr. Zestaw „spójnych, precyzyjnych i zgodnych z obowiązującym prawem przepisów, reguł i procedur, według których dana organizacja buduje, zarządza oraz udostępnia zasoby i systemy informacyjne i informatyczne” [9] to polityka bezpieczeństwa (ang. *security policy*). W zakresie opracowanej i przyjętej polityki bezpieczeństwa określa się co chronić (np. ludzi, sprzęt, dane, systemy, technologie, środowisko), kiedy, jak i kto ma chronić oraz, które zasoby są krytyczne i w związku z tym powinny być szczególnie chronione. Polityka dobrze opracowana, audytowana i udoskonalana pomaga w bezpiecznym funkcjonowaniu przedsiębiorstwa i zachowaniu ciągłości pracy, jeśli jest to konieczne.

* Instytut Informatyki, Wydział Informatyki i Zarządzania Politechniki Wrocławskiej, 50-370 Wrocław, Wybrzeże Wyspiańskiego 27, {hanna.mazur, zygmun.mazur}@pwr.wroc.pl

Niezwykle istotną rolę w zapewnieniu bezpieczeństwa systemów teleinformatycznych odgrywa zarządzanie wiedzą (ang. *knowledge management*), które powinno objąć cały personel instytucji (a często również osoby spoza instytucji – kontrahentów, klientów).

Dawniej majątek ludzi stanowiły pola i fabryki, które wymagały odpowiedniego zarządzania nimi. Dzisiaj dobra, które stanowią ogromną wartość, które muszą być odpowiednio gromadzone, zarządzane i chronione to dane, informacje i wiedza. Dane to fakty, które można przedstawić w postaci par (nazwa, wartość), na przykład: (st_znajomości, „dobry”), (staż, 15). Same w sobie są niezrozumiałe. Nabierają znaczenia po ich zinterpretowaniu, przetworzeniu, dzięki czemu zostaje uwidocznione ich znaczenie, np. stopień znajomości języka niemieckiego przez pracownika jest dobry. Informacje można przetwarzać na nieograniczoną liczbę sposobów i przedstawiać w różnych postaciach (np. jako tekst, graficznie, na wykresie, w formie prezentacji multimedialnej, na ekranie komputera lub wydrukowane na papierze). Wiedza to między innymi zbiór reguł, warunków, doświadczeń, umiejętności pozwalających na poprawne działanie oraz umożliwiających wyciąganie wniosków, przeprowadzanie analiz, stawianie diagnoz i podejmowanie decyzji na podstawie dostępnych informacji. Wiedza to ogół wiarygodnych informacji oraz umiejętność ich wykorzystania. Kilka osób na podstawie tych samych danych może wyciągnąć różne wnioski ponieważ ich dotychczasowa wiedza i dotychczasowe doświadczenia są różne. Wartość 15 trudno właściwie zinterpretować, dana (staż, 15) stanie się informacją, jeśli będziemy wiedzieli, że oznacza staż pracy pracownika naszego zespołu Adama Adamusa. Na podstawie tej informacji i znajomości przepisu, że pracownikowi z 15-letnim stażem pracy przysługuje nagroda jubileuszowa, wiemy, że pan Adamus otrzymał nagrodę finansową i możemy poprosić go o zwrot pożyczonych od nas pieniędzy.

Obecnie ok. 90% informacji powstaje w sposób elektroniczny a 70% tych informacji nie jest drukowana. Wykorzystywanie nowoczesnych technologii informacyjnych w procesach biznesowych organizacji sprawia, że wdrażanie i rozwijanie systemów zarządzania wiedzą oraz polityki bezpieczeństwa staje się koniecznością.

2. SYSTEM ZARZĄDZANIA WIEDZĄ

System zarządzania wiedzą (SZW, ang. *knowledge management system*) wspomaga koordynację pracy w firmie oraz służy do gromadzenia, modyfikacji, porządkowania oraz udostępniania wykorzystywanej i wytwarzanej wiedzy. Ze względu na różnicowane role użytkowników tego systemu (aktorów) często wprowadza się różne uprawnienia, hasła dostępu, funkcje i odpowiedzialności. Udostępniane raporty, analizy, wizualizacje danych oraz różnorodne zestawienia statystyczne przedstawiane w czytelnej postaci (tekstowej, graficznej lub multimedialnej), ułatwiają proces zarządzania, wnioskowania i podejmowania decyzji a także wspomagają prowadzenie polityki bezpieczeństwa.

System zarządzania wiedzą obejmuje strukturę organizacyjną, planowane działania, zakresy odpowiedzialności, obowiązujące zasady i procedury, zachodzące procesy i wykorzystywane zasoby. Aby zapewnić jego poprawne działanie, system musi być monitorowany i weryfikowany (w celu wykrywania nieprawidłowości w funkcjonowaniu, identyfikowaniu incydentów i naruszeń bezpieczeństwa, określania jego skuteczności) oraz modyfikowany zgodnie ze zmianami zachodzącymi w firmie i zgodnie z wynikami monitorowania i weryfikowania systemu oraz ze zgłaszanymi sugestiami pracowników.

Istotnym elementem systemu zarządzania wiedzą jest określenie, co stanowi wiedzę organizacji czyli co gromadzić i dokumentować, w jakim celu, w jaki sposób i w jakiej formie, jak często, kto ma to robić, komu udostępniać dane (które dane, jakie operacje może na nich wykonywać) a także kto ponosi odpowiedzialność za gromadzenie, udostępnianie i modyfikowanie zgromadzonych zasobów oraz jakie są tego koszty i jakie ryzyka.

Nadmiar danych i dokumentów prowadzi często do dezorientacji, powoduje chaos informacyjny, utrudnia i wydłuża czas dostępu do właściwej wiedzy a czasami wręcz uniemożliwia korzystanie z niej. Przechowywanie, na przykład, wielu wersji tego samego dokumentu bez czytelnych adnotacji, utrudnia (a często wręcz uniemożliwia) ustalenie wersji aktualnej.

2.1. CZYNNIKI ODDZIAŁYWAJĄCE NA ZARZĄDZANIE WIEDZĄ

Szybki rozwój Internetu oraz technologii informacyjnych (w tym mobilnych) sprawia, że z jednej strony dostęp do informacji staje się coraz łatwiejszy ale z drugiej strony otrzymujemy czasami zbyt wiele informacji, uzyskane informacje są nieprawdziwe lub sprzeczne. Dezinformacja utrudnia zarządzanie wiedzą. W nadmiarze informacji znalezienie tej odpowiedniej jest często bardzo trudne.

Omawiając zagadnienie zarządzania wiedzą należy uwzględnić zmiany zachodzące w otoczeniu, przeobrażenia ku społeczeństwu informacyjnemu (co oznacza coraz więcej ludzi zatrudnionych przy przetwarzaniu informacji oraz swobodny dostęp do technologii informacyjnych), zmiany współczynnika skolaryzacji (powszechności nauczania) społeczeństwa, tempo zmian technologicznych i kadrowych, postępującą globalizację oraz internacjonalizację przedsiębiorstw. Gwałtowny wzrost międzynarodowej wymiany handlowej i kulturowej, zmiany w przepisach podatkowych, wejście do Unii Europejskiej, umiędzynarodowienie działalności firm, zmiana profilu firm i przebranżowienie w wielu dziedzinach gospodarki oznacza potrzebę wypracowania nowych procedur i dokumentów, nowego systemu zarządzania wiedzą, nowej polityki bezpieczeństwa, dodatkowych szkoleń pracowników itd. Częste i obejmujące dużą grupę pracowników zmiany kadrowe mogą mieć wpływ na system zarządzania wiedzą. Jest to odczuwalne

również w oświacie i na wyższych uczelniach. Z kolei przedsiębiorstwo dotychczas bardzo stabilne kadrowo może odczuć potrzebę wdrożenia systemu zarządzania wiedzą po wieloletniej działalności np. z powodu nagłego odejścia wielu wykwalifikowanych pracowników na emeryturę. Firma, która nie dbała o zachowanie ciągłości kadrowej może stanąć przed problemem zapewnienia dotychczasowego poziomu prowadzonej działalności.

2.2. PROBLEMY W PRZEKAZYWANIU WIEDZY

Problemy w przekazywaniu wiedzy wynikają z różnych przyczyn. W celu właściwego zarządzania wiedzą muszą być opracowane pewne procedury i normy postępowania, które muszą być spisane i muszą obejmować wszystkie sytuacje. Dlatego najpierw musi być przeprowadzona dokładna analiza, studium przypadku (ang. *case study*). Zazwyczaj opracowane dokumenty są obszerne i nużące, często modyfikowane (np. przy zmianach zachodzących w przedsiębiorstwie) i wymagają dużo czasu na ich przeczytanie i zapamiętanie. Na ogół pracownik przywiązuje wagę do tych zaleceń, które dotyczą sytuacji typowych, powtarzalnych i związanych z jego zakresem obowiązków. Tymczasem do problemów dochodzi często w przypadkach nietypowych lub awaryjnych.

Oczywiście przepływ wiedzy może odbywać się w sposób samoistny lub przypadkowy ale najlepiej gdy jest zorganizowany i kontrolowany.

W wielu przedsiębiorstwach, oprócz zasad i reguł spisanych, funkcjonują pewne reguły niepisane, zwyczajowo przyjęte i akceptowane. Taki stan często odpowiada pracownikom i ich przełożonym, gdyż dopuszcza pewną swobodę działania i postępowanie według zmiennych kryteriów. Są oni zatem przeciwni wprowadzeniu systemu zarządzania wiedzą i uporządkowaniu pewnych procesów.

Niezwykle ważne jest szybkie reagowanie na zachodzące zmiany. Bywa, że zostają wypracowane nowe schematy pracy i pracownicy zauważają konieczność dostosowania dotychczasowych dokumentów i procedur postępowania do nowych warunków i okoliczności ale nie zawsze od razu przekazują stosowne informacje do organów nadrzędnych, a te z kolei, nie zawsze odpowiednio reagują na sygnalizowane potrzeby zmian.

Pojawianie się np. nowych form komunikacji czy nowych nośników danych, wzrastająca konkurencyjność firm wymuszają dostosowywanie przekazywania wiedzy do nowych warunków, podnoszenie efektywności w przekazywaniu informacji, zwiększanie tempa zmian w tym zakresie i ciągle wprowadzanie innowacji.

Na decyzję firmy co do wdrożenia systemu zarządzania wiedzą w formie elektronicznej ma wpływ konieczność poniesienia dodatkowych kosztów, obawa o niezgodność oferowanych funkcji systemu z wymaganiami przedsiębiorstwa (teraz lub w przyszłości) oraz o dostęp do uaktualnień systemu, obawy związane z utrzymywa-

niem systemu i ze skalowalnością systemu (w przypadku rozbudowy firmy, zwiększenia liczby użytkowników, dokumentów, usług), konieczność zapewnienia ciągłości pracy w przypadku zmian, awarii lub potrzeby wymiany czy rozbudowy sprzętu lub oprogramowania.

Przykładowymi barierami w wymianie wiedzy są:

- brak odpowiedniej infrastruktury komunikacyjnej,
- wysokie koszty zdobywania nowej wiedzy i pozyskiwania wykwalifikowanych pracowników,
- celowe utajnianie i nie przekazywanie pełnej wiedzy współpracownikom np. w celu utrzymania przewagi (władzy),
- brak ciągłości kadrowej, zmiany kadrowe,
- odrzucanie (lekceważenie) pomysłów danego pracownika lub pewnej grupy osób,
- brak pozwolenia (akceptacji) kierownictwa na zmiany i nowe pomysły/prace, blokowanie zgłaszanych innowacji,
- brak pracy zespołowej, konkurencyjność działania pracowników,
- brak odpowiedniego systemu motywacyjnego lub zły system motywacyjny,
- brak opracowanych procedur regulujących wymianę informacji,
- brak kontroli przekazywania informacji,
- trudności w ustaleniu winnych nie przekazywania informacji,
- brak więzi pomiędzy pracownikami,
- brak otwartości na dobre pomysły pracowników organizacji,
- brak informacji na temat wiedzy posiadanej przez innych pracowników.

3. GŁÓWNE ZADANIA SYSTEMU ZARZĄDZANIA WIEDZĄ

O systemach zarządzania wiedzą mówi się od lat dziewięćdziesiątych XX wieku, chociaż już wcześniej istniały np. grupy dyskusyjne w Internecie czy repozytoria dokumentów i przypadków. Badania przeprowadzone w 1999 roku wykazały, że w 69% przypadków SZW pojawiły się samoistnie wraz z rozwojem infrastruktury IT w organizacji [KPMG 1999]. Tak więc SZW funkcjonowały wcześniej a dopiero w latach 90. XX wieku nadano im nazwę i ujęto w pewne ramy.

Główne zadania systemów zarządzania wiedzą to transfer i weryfikacja wiedzy oraz innowacyjność. Zagadnienia te zostaną przedstawione w kolejnych podpunktach.

3.1. TRANSFER WIEDZY

W ramach transferu wiedzy w firmie należy odpowiednio szybko przekazywać niezbędną wiedzę pracownikom a następnie, jeśli to konieczne, weryfikować, czy

została odebrana i w dostatecznym stopniu przyswojona. Rozpatrując zagadnienie transferu wiedzy należy uwzględnić jej różne rodzaje:

- *know how* – umiejętność i doświadczenie – wiedza specyficzna dla każdego przedsiębiorstwa, budowana przez wiele lat, decyduje o konkurencyjności firmy i jej przewadze rynkowej;
- *know what* – wiedza o faktach, analizach, dotyczy definicji pojęć i opisów zrozumiałych dla członków danej grupy, dzięki stosowaniu specyficznego (specjalistycznego) słownictwa;
- *know why* – wiedza o zasadach i regułach;
- *know who* – wiedza o ludziach, ich kompetencjach, umiejętnościach, predyspozycjach itd.

Zróznicowany charakter wiedzy ma wpływ na różnorodność jej transferu, który może być:

- seryjny – wykorzystywany przez zespoły do wykonywania podobnych zadań, bazuje na wcześniejszych doświadczeniach zespołu, pozwala na uniknięcie kosztownych błędów oraz na wzrost efektywności, wydajności, skuteczności i jakości prowadzonej działalności;
- wierny – polegający na przekazaniu przez członków jednego zespołu doświadczeń i wiedzy, czyli tzw. najlepszych praktyk (ang. *best practices*), członkom innego zespołu, wykonującemu podobne zadania w innym terytorialnie miejscu;
- odległy (naśladowczy) – stosowany na ogół przy wykonywaniu zadań specjalnych, nietypowych i wykorzystujący wiedzę i doświadczenie osób z innego zespołu (dzięki znajomości poszczególnych członków zespołu);
- strategiczny – dotyczy na ogół zadań istotnych i złożonych, o bardzo dużym znaczeniu dla całokształtu działalności przedsiębiorstwa;
- ekspercki – związany z bardzo specjalistycznymi dziedzinami, najczęściej w postaci trudnych do zrozumienia (ze względu na użyte specjalistyczne słownictwo) ekspertyz i opracowań, sporządzonych przez ekspertów i biegłych.

3.2. WERYFIKACJA WIEDZY

Zarządzanie wiedzą wymaga ciągłego weryfikowania i podnoszenia standardów oraz dostosowywania wypracowanych procedur.

Prawidłowe zarządzanie wiedzą wymaga ustalenia sposobu dokumentowania, miejsca przechowywania dokumentów, określenia osób upoważnionych do korzystania i modyfikowania dokumentów. Ważnym elementem jest również informacja o zakresach obowiązków i kompetencjach członków zespołu, w celu właściwego funkcjonowania firmy. Brak znajomości tych informacji może spowodować przekazanie haseł lub informacji poufnych osobom nieupoważnionym lub niewłaściwe przypisanie personelu do wykonywania zadań.

3.3. INNOWACYJNOŚĆ

We współczesnej gospodarce szczególnego znaczenia nabiera innowacyjność. Oparcie działalności przedsiębiorstwa na wiedzy, innowacyjności, nowych technologiach i badaniach naukowych umożliwia rozwój i postęp oraz osiąganie sukcesów, a dzięki temu, konkurencyjność w stosunku do innych firm z danej branży.

Innowacyjność (ang. *innovation*) czyli nowatorstwo oznacza wprowadzanie nowych elementów, rozwiązań, technologii, w celu ulepszenia i poprawienia warunków pracy oraz osiąganych wyników. Może dotyczyć zarówno produktów jak i procesów przemysłowych. Niestety, wiele innowacyjnych rozwiązań nie zostaje właściwie wykorzystywanych, między innymi, ze względów finansowych oraz z braku odpowiedniego zarządzania wiedzą – brak jest skorelowania innowacyjnych rozwiązań z potencjalnymi zainteresowanymi i sponsorami finansowymi.

Innowacja oznacza pewne zmiany (czasami dość znaczne), nie zawsze przez wszystkich pożądanę. Powodem blokowania innowacyjności i wprowadzania nowatorskich rozwiązań jest lęk pracowników przed reorganizacją, obawa o utratę swojej pozycji czy stanowiska, co w konsekwencji doprowadza do powstawania grup blokujących wprowadzanie zmian i innowacyjnych rozwiązań. W takim przypadku istotną rolę mogą odegrać spotkania kadry zarządzającej z pracownikami i odpowiedni przepływ informacji.

4. POLITYKA BEZPIECZEŃSTWA

Opracowując politykę bezpieczeństwa firmy (w tym politykę bezpieczeństwa systemów teleinformatycznych) należy zdefiniować podstawowe pojęcia, dokonać klasyfikacji oraz kontroli posiadanych i chronionych zasobów (materiałowych, ludzkich), określić grupy użytkowników i ich role a także sposoby dostępu do systemów i danych, opracować zasady eksploatacji i konserwacji systemów teleinformatycznych, dokonać selekcji procesów biznesowych wymagających bezwarunkowej ciągłości i określić sposoby zapewnienia takiej ciągłości, zaplanować szkolenia pracowników, dokonać analizy ryzyka związanego z awarią, oszacować koszty zaplanowanej i ewentualnie przyjętej polityki bezpieczeństwa oraz koszty jej nie przyjęcia. Wynikiem prowadzonej polityki ma być zapewnienie bezpieczeństwa zasobów (sprzętu, systemów, danych, pracowników, środowiska) i prawidłowego działania firmy. Brak polityki bezpieczeństwa naraża firmę na utratę danych a tym samym na utratę dobrego imienia co może spowodować nawet bankructwo firmy.

Ze względu na różnorodny charakter firm, prowadzonej działalności oraz odmienny sposób ich funkcjonowania, polityki bezpieczeństwa przedsiębiorstw też mogą się znacznie różnić od siebie. Zbudowanie i przestrzeganie polityki bezpieczeństwa dla danej firmy wymaga ukierunkowanego (dedykowanego) podejścia i za

każdym razem jest to bardzo indywidualny program (i zazwyczaj kosztowny). Decyzję o jego opracowaniu i przyjęciu należy podjąć po dogłębnej analizie potrzeb, ryzyka i kosztów.

Istotnym aspektem w zachowaniu bezpieczeństwa informacji jest na przykład odpowiednie ustalanie haseł dla pracowników i przez pracowników do kont internetowych. Zbyt duża liczba haseł sprawiających kłopot z ich zapamiętaniem powoduje, że pracownicy zapisują je i to najczęściej w miejscach łatwo dostępnych, np. w notesach lub na luźnych kartkach przechowywanych w miejscach łatwo dostępnych (dla nich i często dla innych również). Jak wynika z raportu firmy Cyber-Ark Software około 50% pracowników działów IT przechowuje hasła na kartkach przyklejanych do monitorów [Security 2008]. Część osób przechowuje hasła w plikach na komputerze. Hasła powinny być silne czyli mieć od 8 do 14 znaków z całej klawiatury (liter, cyfr, symboli), łatwe do zapamiętania dla właściciela ale trudne do odgadnięcia dla innych, powinny być okresowo zmieniane i różne do różnych kont, w szczególności inne do kont prywatnych i do służbowych, nie powinny odpowiadać nazwie użytkownika ani zawierać ciągów powtórzeń znaków. W praktyce niewiele osób stosuje się do tych zasad nie wyłączając pracowników działów IT (co trzeci administrator zmienia swoje hasła raz na kwartał, a 9% nie robi tego nigdy). Aż w 82% firm hasła administratorów nie są nigdzie przechowywane (18% firm zapisuje je elektronicznie). Bywa też, że hasło administratora pozostaje nie zmienione po zmianie pracownika na tym stanowisku. Z powyższych danych wynika, że dopóki przedsiębiorstwa nie zaczną stosować odpowiednich zabezpieczeń, SZW oraz odpowiedniej polityki bezpieczeństwa trudno będzie zapewnić ochronę i poufność danych. Konieczne jest zabezpieczanie sieci, wprowadzanie i przestrzeganie zasad przechowywania i przesyłania danych oraz kontrolowanie przywilejów użytkowników.

Dla zapewnienia środków adekwatnych i proporcjonalnych do prowadzonej działalności oraz stanu zasobów przedsiębiorstwa, należy zaprojektować i wdrożyć system zarządzania bezpieczeństwem informacji (SZBI). Z polskiej normy PN-ISO/IEC 27001:2007 wynika, że każda organizacja powinna „ustanowić, wdrożyć, eksploatować, monitorować, przeglądać, utrzymywać i stale doskonalić udokumentowany SZBI w kontekście prowadzonej przez nią działalności i ryzyka występującego w organizacji.” [PN 27001].

Wytyczne dotyczące zarządzania bezpieczeństwem informacji (ang. *information security management*) zawarte są w:

- dokumencie ISO/IEC TR 13335,
- normie PN-ISO/IEC 27001 (która zastępuje normę ISO/IEC 17799, *Code of practice for information security management* oraz odpowiadający jej standard brytyjski BS 7799-2),
- dokumentach NIST (ang. *National Institute of Standards and Technology*),
- zaleceniach COBIT (ang. *Control Objectives for Information and Related Technology* – Cele Kontrolne dla Technologii Informacyjnej i Technologii

Związanych) dla wytyczenia zakresu kontroli procesów informatycznych obsługujących biznes.

Normą dla zabezpieczenia połączeń zewnętrznych w sieciach teleinformatycznych jest dokument ISO/IEC TR 13335-5:2004 *Information technology – Guidelines for the management of IT Security Part 5: Management Guidance on network security*. Ważnym dokumentem z zakresu bezpieczeństwa sieciowego jest standard ISO/IEC 18028 *Information technology – Security techniques – IT network security*, części 1–5.

Normy ISO/IEC 18028 są źródłem informacji dotyczących bezpieczeństwa sieciowego oraz zawierają wzorce procedur i dokumentów.

Raport ISO/IEC TR 15947:2002 dotyczy zagadnień wykrywania włamań do systemów teleinformatycznych i zawiera wytyczne odnoszące się do architektury i funkcjonalności systemów monitorowania bezpieczeństwa.

Norma ISO/IEC 18043 (z czerwca 2006 roku) *Information technology – Security techniques – Selection, deployment and operations of intrusion detection systems* dotyczy systemów wykrywania włamań (ang. *Intrusion Detection System*, IDS). Obszerne omówienie zagadnień dotyczących bezpieczeństwa informacji i usług jest na przykład w książce [Białas 2006].

5. POLITYKA BEZPIECZEŃSTWA A ZARZĄDZANIE WIEDZĄ

Celem zarządzania wiedzą jest wspomaganie określonej działalności w obszarach, w których niezbędne jest gromadzenie, przekazywanie oraz wykorzystywanie wiedzy. Obejmuje między innymi dokumentowanie bieżącej działalności, przechowywanie, udostępnianie i archiwizowanie danych, przekazywanie wiedzy oraz obieg informacji.

Zarządzanie wiedzą, związaną z zapewnieniem bezpieczeństwa systemów teleinformatycznych (jej dokumentowaniem, przechowywaniem, udostępnianiem, obiegiem, archiwizowaniem, drukowaniem, kopiowaniem, usuwaniem itd.), jest sprawą niezwykle istotną i absolutnie nie można tego aspektu pominąć. Tymczasem, jak wynika z udostępnianych wyników badań, w wielu przypadkach traktowane jest marginalnie i jest przyczyną wielu problemów. W firmie muszą zostać wypracowane procedury dotyczące pracowników w kontekście ich wiedzy, postępów w pracy, osiągniętych wyników i spostrzeżeń, podejmowanych i realizowanych zadań. Muszą być wypracowane procedury i zalecenia odnośnie gromadzenia, przechowywania, przekazywania oraz dostępności informacji kluczowych dla działalności systemów. Pewne dane z jednej strony muszą być z pewnością chronione, a z drugiej strony, muszą być dostępne (i odpowiednio zabezpieczone) w przypadku nieprzewidzianych zdarzeń (nagłe odejście pracownika z firmy, nagły wypadek, choroba). Z raportu firmy Tower Software za 2007 rok wynika, że 55% ankietowanych pracowników brytyjskich przechowuje dane firmowe poza współdzieloną siecią komputerową.

Wiedza, jaką dysponują pracownicy, to zarówno wiedza formalna, uzewnętrzniana w pewien sposób np. mówiony lub pisemny (spisane procedury postępowania, wytyczne, instrukcje postępowania, schematy organizacyjne) jak i wiedza cicha (ukryta), wykorzystywana przez pracowników ale nie wyrażona w sposób formalny, często bardzo trudna do sformalizowania. Wiedza cicha jest oczywista dla dysponującego nią pracownika ale często trudna do przekazania innym lub celowo zatajana np. dla podniesienia swojego prestiżu. Popularnie określana jest jako *know-how* (wiedza aksjomatyczna) i *know-who* (wiedza relacyjna).

To oczywiste z pozoru zadanie dotyczące dokumentowania istotnej wiedzy często nie jest wykonywane należycie z różnych powodów: skomplikowania i złożoności, braku czasu, braku kontroli przez organy nadrzędne, nadmiernego optymizmu pracownika lub braku świadomości o strategicznym znaczeniu tej czynności, często zmieniającego się stanu wiedzy, co powoduje przesuwanie w czasie momentu jej udokumentowania, celowe działanie w celu podniesienia swojej pozycji w firmie itp. Ułatwieniem dla firm w zakresie zarządzania wiedzą mogą być dostępne programy np. Empolis, Pyton Management, które wspomagają wykonywanie kluczowych procesów biznesowych i organizacyjnych, zarządzanie dużymi zasobami dokumentów, zaawansowane zarządzanie treścią (metainformacjami, strukturą, powiązaniem, uprawnieniami), przepływem pracy, inteligentne wyszukiwanie i modelowanie wiedzy.

Coraz większego znaczenia nabiera praca zespołowa i komunikacja między członkami zespołu. Ciekawe spostrzeżenia na temat komunikacji w zespole zawarte są w [Security 2008].

Kolejny istotny aspekt, to podnoszenie kwalifikacji i umiejętności pracowników poprzez ich samodoszktałcanie się oraz organizowanie szkoleń w zakresie bezpieczeństwa systemów. Pracodawca musi uwzględnić w harmonogramie pracy czas przeznaczony na doszktałcanie się, gdyż, przy burzliwym rozwoju technologii informacyjnych, jest to konieczność. Z pewnością pracownik uczy się w trakcie wykonywania pracy na bieżąco: z chwilą napotkania problemu czy nowego zadania. Jednak nauka taka jest bardzo wybiórcza, niepełna i wycinkowa a zdobywania wiedzy w ten sposób zajmuje nieraz o wiele więcej czasu niż ukierunkowane, prowadzone przez specjalistów szkolenia. Ponadto, być może wielu pracowników napotyka te same problemy i zupełnie niepotrzebnie przecierają te same szlaki w ich rozwiązywaniu. W różnych dziedzinach może być różne zapotrzebowanie na szkolenia, mniejsze, na przykład, jeśli kadrę stanowią doświadczeni i merytorycznie bardzo dobrze przygotowani pracownicy. Jednak w przypadku personelu związanego z polityką bezpieczeństwa, ze względu na charakter zagadnień, szkolenia i ciągle doszktałcanie się są elementami niezbędnymi. Do kompleksowej obsługi organizacji w zakresie zarządzania i wspomagania rozwojem kapitału ludzkiego wykorzystywane są systemy HCDM (*Human Capital Development & Management* – Zarządzanie i Rozwój Kapitału Ludzkiego), których największym dostawcą jest firma Saba Software (według rankingu Fortune 500 aż 60% największych firm na świecie używa platformy Saba do wspomagania procesu zarządzania wiedzą).

5.1. DOKUMENTOWANIE POLITYKI BEZPIECZEŃSTWA A ZARZĄDZANIE WIEDZĄ

Utrzymanie bezpieczeństwa informacji oraz środków technicznych wykorzystywanych do zarządzania wiedzą wiąże się z odpowiednią organizacją wewnątrz przedsiębiorstwa jak i z określeniem procedur i zabezpieczeń w kontaktach ze stronami zewnętrznymi (innymi firmami, klientami, partnerami), z właściwym zarządzaniem aktywami (czyli zasobami mającymi wartość dla organizacji) np. w przypadku rzeczy materialnych poprzez ich identyfikację i systematyczne inwentaryzowanie, określenie właściciela aktywu i miejsca przechowywania, określenie, zdokumentowanie i przestrzeganie zasad korzystania z zasobów.

Wszystkie postanowienia i procedury muszą być opracowane i przechowywane w formie pisemnej.

Kandydaci na pracowników muszą być zweryfikowani pod kątem kompetencji zawodowych, moralnych, prawnych oraz przydatności do powierzanych im w przyszłości zadań. Osoby zatrudnione muszą mieć ściśle określony zakres praw i obowiązków, muszą je rozumieć i przestrzegać oraz znać konsekwencje naruszenia bezpieczeństwa informacji. W przypadku koniecznych zmian w polityce bezpieczeństwa lub w zarządzaniu wiedzą muszą być o tym powiadamiani i, jeśli zachodzi taka potrzeba, to przeszkoleni.

Przyjęta polityka bezpieczeństwa informacji oraz system zarządzania wiedzą muszą szczegółowo precyzować sposób postępowania pracownika w przypadku zmiany stanowiska lub planowanego odejścia z firmy a więc jakie aktywa ma zwrócić, w jakim terminie i w jakiej formie, kiedy należy mu odebrać prawa dostępu do informacji, usunąć jego konto pocztowe itp. Około 25% ankietowanych przez firmę Cyber-Ark przyznało, że zwolniony pracownik w ich zakładzie pracy jeszcze przez wiele miesięcy ma dostęp do zasobów firmy [Security 2008].

Właściwe zarządzanie wiedzą wymaga właściwego dokumentowania zgodnego z przyjętą polityką bezpieczeństwa. Muszą zostać sprecyzowane wszelkie wymagania dotyczące dokumentacji a w szczególności, co dokumentacja powinna obejmować oraz uregulowania dotyczące nadzoru nad dokumentami i zapisami. Wymagania takie zawarte są zarówno w polskiej normie PN-ISO/IEC 27001, jak i w normach międzynarodowych ISO 9001:2000 oraz w ISO 14001:2004.

5.2. ŹRÓDŁA ZAGROZEŃ DLA BEZPIECZEŃSTWA TELEINFORMATYCZNEGO

Niemal każde większe przedsiębiorstwo korzysta z nowych technologii teleinformatycznych w tym z komputerów podłączonych do sieci Internet. Często dostęp do Internetu jest koniecznością dla ciągłej i sprawnej pracy ale jednocześnie stwarza

ogromne zagrożenie dla bezpieczeństwa danych i musi być uwzględniony w zarządzaniu wiedzą i polityce bezpieczeństwa informacji.

Powszechne zagrożenia dla bezpieczeństwa danych ludzie wymieniają bez większych problemów, np. pochodzące z sieci Internet wirusy, robaki, konie trojańskie, spam, zainfekowane maile itd. Tymczasem, przeprowadzane badania wykazują, że najwięcej zagrożeń spowodowanych jest przez pracowników firmy.

Jak wynika z raportu *Computer Crime and Security Survey for 2007* amerykańskiego Computer Security Institute, opracowanego na podstawie ankiet wypełnionych przez ok. 500 specjalistów do spraw bezpieczeństwa, aż 59% ankietowanych stwierdziło, że w ciągu ostatniego roku w ich przedsiębiorstwie doszło do incydentu zagrażającego bezpieczeństwu informatycznemu, który spowodowany był przez pracownika danej firmy (poprzez kradzież danych, nieświadome udostępnianie danych ale również przez przeglądanie zainfekowanych stron internetowych czy korzystanie z nielegalnego, zawirusowanego oprogramowania). W tym samym czasie natomiast zagrożenie ze strony szkodliwego oprogramowania, w postaci wirusa, robaka czy konia trojańskiego, odnotowano w 52% przypadków [www 2008].

Dużym zagrożeniem dla bezpieczeństwa danych są (niestety) administratorzy, którzy ujawniają poufne dane z chęci zysku, z zemsty lub z bezmyślności.

W wielu zakładach pracownicy mają zakaz instalowania na komputerach firmowych własnego oprogramowania, często nielegalnego i skopiowanego z sieci Internet. Podczas instalacji takiego oprogramowania jednocześnie często jest instalowane (bez wiedzy i zgody użytkownika) oprogramowanie szkodliwe (malware), szpiegujące (spyware) lub przechwytyjące naciskane na klawiaturze klawisze (keylogery). Według firmy IDC (*International Data Corporation*) wejście na stronę z nielegalnym oprogramowaniem oznacza 25% zagrożenie jednoczesnego ściągnięcia złośliwego kodu.

Z badań przeprowadzonych w USA w marcu 2008 roku przez firmę Applied Research-West na zlecenie firmy SanDisk wynika, że aż 77% ankietowanych (telefonicznie) pracowników wykorzystuje pamięć USB, głównie do kopiowania danych klientów (25%), danych finansowych (17%), planów biznesowych (15%), danych pracowników (13%), planów marketingowych (13%), własności intelektualnej (6%) i kodów źródłowych (6%), przy czym kadra kierownicza nie jest tego świadoma i ocenia skalę tego zjawiska na ok. 35% [Sandisk 2008]. Pracownicy kopią dane na pamięci Flash pomimo świadomości dużego ryzyka utraty poufności danych i wynikających z tego tytułu strat. Chociaż 67% ankietowanych firm wdrożyło lub wdraża politykę bezpieczeństwa, to nadal 23% użytkowników nie zna zasad dotyczących użytkowania urządzeń USB. Ankietowani reprezentują bardzo różny poziom wiedzy o prowadzonej polityce bezpieczeństwa w zakładzie pracy i aż 44% w ogóle nie wie, że firma ma opracowaną politykę bezpieczeństwa zabraniającą kopiowania danych na PenDrive. Pomimo prowadzonych szkoleń w zakresie zasad użytkowania pamięci USB (33% ankietowanych miało szkolenie raz w roku, 24% kilka razy w roku, 22% – raz przy przyjmowaniu do pracy, 17% – w miarę potrzeby, 3% – ani razu), to zasady

te są w pełni zrozumiałe tylko dla 19% ankietowanych, a dla 21% zrozumiałe jedynie w ograniczonym zakresie.

Kolejne niepokojące zjawiska odnotowane w 2007 roku, to coraz częstsze (proporcjonalnie do wzrostu ich popularności) zaginięcia (kradzieże) laptopów, palmtopów (*Personal Digital Assistant* – PDA) oraz smartphone’ów (przenośnych urządzeń oferujących usługi telefonu, poczty elektronicznej, przeglądarki internetowej, aparatu cyfrowego, kamery video) wraz ze zgromadzonymi danymi.

Zasoby firm w postaci danych, dokumentów, gromadzonej wiedzy są cennym towarem. Przeprowadzane na nie ataki pasywne polegają na monitorowaniu i przechwytywaniu tych zasobów, natomiast ataki aktywne – na ingerencji w te zasoby, na przykład poprzez ich zmianę, usunięcie, zniszczenie. Zdobywanie dostępu do chronionych zasobów przedsiębiorstwa możliwe jest, między innymi, dzięki wykorzystaniu inżynierii społecznej (ang. *social engineering*) czyli ataku socjotechnicznego.

5.3. ATA K SOCJOTECHNICZNY

Atak socjotechniczny polega na uzyskiwaniu informacji poprzez odpowiednie manipulowanie ludźmi a więc na pozyskaniu zaufania wybranej osoby i dzięki temu uzyskaniu określonych danych lub dostępu do wybranych czynności czy zasobów. W tym celu wykorzystywane są różne techniki manipulacji i podstępu, najczęściej bazujące na łatwowierności i naiwności ludzi ale również na nieprzestrzeganiu procedur, chęci bycia pomocnym lub osiągnięcia dodatkowych korzyści (niekoniecznie materialnych ale np. sympatii innej osoby, zainteresowania sobą, dowartościowania swojej pozycji itd.). Kevin Mitnick (włamywacz komputerowy) w książce [Mitnick 2003] opisał wiele technik inżynierii społecznej oraz metod ochrony przed tego typu atakami. Mitnick przedstawił niski stan zabezpieczeń sieci teleinformatycznych w USA i złudne poczucie ich właściwej ochrony. Przytoczył wiele przykładów wskazujących, że w najlepiej zabezpieczonych systemach najsłabszym ogniwem był człowiek. Dlatego tak ważna jest profilaktyka w postaci podnoszenia świadomości pracowników oraz wprowadzania i egzekwowania odpowiednich procedur postępowania oraz szybkiego reagowania. Nie istnieją bowiem żadne zabezpieczenia techniczne stuprocentowo wykluczające powodzenie technik socjotechnicznych. Sposobem na ich ograniczanie jest np. monitorowanie pracy pracownika, jawne nagrywanie rozmów telefonicznych, przydzielanie kilku pracowników do tych samych zadań czy pomieszczeń, szkolenia pracowników. Wszystkie te sposoby nie stanowią jednak wystarczających zabezpieczeń, czego przykładem są nadal z powodzeniem przeprowadzane tego typu ataki.

Rodzajem ataku socjotechnicznego jest phishing, którego celem jest wydobycie cennych informacji osobistych takich jak hasło dostępu czy numer karty kredytowej poprzez podszywanie się pod osobę (instytucję) godną zaufania, której te informacje są pilnie potrzebne. Czasami wystarczy jeden błąd pracownika, który umożliwi oszustomu uzyskanie poufnych danych firmy. Według badań firmy Gatner w okresie

08.2006–08.2007 roku, ofiarami phishingu było ponad 3,5 mln mieszkańców USA, przy czym zestawienie to uwzględnia tylko osoby, które w wyniku działalności przestępców poniosły straty finansowe (średnio każdy okradziony stracił około 900 dolarów).

W Polsce stosowanie ataków socjotechnicznych jest przestępstwem. Instytucje publiczne oraz firmy dopuszczające kontakty elektroniczne pracowników i klientów muszą wprowadzać systemy autoryzacji. Pracownicy posiadający dostęp do informacji niejawnych, wrażliwych i szczególnie chronionych stale muszą podnosić swoje kwalifikacje w zakresie zasad bezpieczeństwa teleinformatycznego.

Ocena podatności pracowników na ataki socjotechnicznych możliwa jest, na przykład, poprzez przeprowadzanie audytów bezpieczeństwa.

6. AUDYTY BEZPIECZEŃSTWA

Audyt (ang. *audit*) to niezależna i udokumentowana ocena zgodności stanu faktycznego z obowiązującymi standardami, przepisami i udokumentowanymi wymaganiami. Audyt może być wewnętrzny (przeprowadzony przez audytora wewnętrznego czyli pracownika firmy) lub zewnętrzny (przeprowadzany przez audytorów zewnętrznych). Audyt musi być zrealizowany rzetelnie i zgodnie z harmonogramem. Tytuł audytora jest przyznawany tylko na trzy lata i aby go utrzymać, trzeba udowodnić stały rozwój zawodowy. W Polsce w jednostkach sektora publicznego musi być przeprowadzany audyt finansowy (na podstawie Ustawy o finansach publicznych z dnia 30 czerwca 2005 r. [10]) ale nie ma jednoznacznych przepisów prawnych nakładających obowiązek przeprowadzenia audytu informatycznego.

Audyt bezpieczeństwa (ang. *security audit*) systemu teleinformatycznego polega na ocenie działania systemu i zastosowanych zabezpieczeń, ich zgodności z przyjętą polityką bezpieczeństwa, określeniu braków w zabezpieczeniach i możliwych zagrożeniach (np. ze strony osób nieuprawnionych lub czynników niezależnych, takich jak awaria urządzeń czy zjawiska atmosferyczne) oraz na wskazaniu rozbieżności z założonymi wymaganiami.

Wykonanie audytu wiąże się z dodatkowymi kosztami i oddelegowaniem pracowników do współpracy z zespołem audytowym, dlatego decyzja o konieczności wykonania audytu powinna wynikać z profilu audytowanej jednostki, kompetencji zatrudnionych pracowników, wniosków i zaleceń poprzednio wykonanego audytu itd. Terminy wykonywania audytów ustala się biorąc pod uwagę między innymi wprowadzane zmiany, zaistniałe sytuacje krytyczne, zalecenia z poprzedniego audytu. Wyniki z przeprowadzonego audytu powinny być przedstawione w uzgodnionej formie np. w postaci raportów zawierających ocenę stanu faktycznego, zaleceń, wytycznych do realizacji, planów na przyszłość i wniosków końcowych, według ustalonych zasad i z podaniem zasad kontroli realizacji wniosków z przeglądu bezpieczeństwa.

W Polsce sprawami audytów zajmuje się niezależne stowarzyszenie ISACA (ang. *Information Systems Audit and Control Association* – Stowarzyszenie do spraw Audytu i Kontroli Systemów Informatycznych), które jest największą i najważniejszą organizacją zajmującą się problemami audytu, kontroli i zarządzania w środowisku informatycznym. Najpopularniejsze prace ISACA to: COBIT – standard zarządzania i kontroli systemów informatycznych, CONeCT – standard audytu środowiska sieciowego oraz program edukacyjny PSS (ang. *Professional Seminar Series*). Stowarzyszenie prowadzi także ogólnosiwiatowy program certyfikacji audyto-rów systemów informatycznych – CISA (ang. *Certified Information Systems Auditor*).

W wyniku przeprowadzenia audytu bezpieczeństwa sprzętu, oprogramowania i procesów można wskazać słabe punkty w zabezpieczeniach systemów informacyjnych: zarówno legalne, jak i łatwe do przełamania przy zastosowaniu ataku socjotechnicznego.

Aby sprawdzić odporność pracowników firmy na ataki socjotechniczne przeprowadza się audyty bezpieczeństwa procesów polegające, między innymi, na testowaniu zachowań pracowników, ich podatności na przekazywane opinie, stopień przestrzegania procedur, lojalność względem firmy, dyskrecję. W tym celu wykorzystuje się wszelkie możliwe sposoby oddziaływania i drogi kontaktów (listowne, telefoniczne, elektroniczne, towarzyskie), preparowane są nawet nieprawdziwe informacje i sytuacje, na przykład awaryjne, w czasie których próbuje się uzyskać od pracownika informacje poufne, hasła oraz dostęp do chronionych zasobów z pominięciem obowiązujących procedur.

7. PODSUMOWANIE

Systemy zarządzania wiedzą (SZW) ułatwiają zarówno gromadzenie danych, informacji i wiedzy jak i zarządzanie nimi oraz kompetencjami ludzi. Dzięki przydzielonym prawom dostępu i autoryzacji praw umożliwiają koordynację pracy, optymalne wykorzystanie wiedzy, kompetencji i czasu pracowników, monitorowanie przebiegu prac oraz przepływu informacji i dokumentów. Wizualizacja danych oraz różne rodzaje raportów ułatwiają analizę danych i podejmowanie decyzji. Systemy zarządzania wiedzą niewątpliwie wymuszają uporządkowanie, systematyczność i terminowość co pozwala osiągać lepsze wyniki pracy i podnosić konkurencyjność firm. SZW wykorzystują nowoczesne technologie informatyczne i informacyjne, często wykorzystują sieci Intranet i Internet, a w związku z tym niezwykle ważnym elementem jest zapewnienie bezpieczeństwa gromadzonym i przesyłanym danym.

W Polsce, podobnie jak i w innych krajach, coraz większą wagę przywiązuje się do wprowadzania polityki bezpieczeństwa informacji i systemów zarządzania wiedzą. Utrata informacji przez firmę prowadzi do utraty zaufania partnerów biznesowych i zagraża stabilności firmy. Niestety, rozwijające się w szybkim tempie współczesne technologie informacyjne oferują coraz większe możliwości, nie tylko użytkownikom ale również oszustom.

Zarządzanie wiedzą jest ściśle związane z zarządzaniem zmianami i transferem wiedzy oraz motywowaniem pracowników do działań innowacyjnych. Obecnie coraz większego znaczenia nabiera praca zespołowa i komunikacja w zespołach i między zespołami, współpraca międzynarodowa oraz szybki i bezpieczny przepływ informacji.

Polityka bezpieczeństwa informacji musi być zgodna ze wszystkimi innymi wykorzystywanymi systemami, w tym, z systemem zarządzania wiedzą. Musi być tak opracowana by jej stosowanie i przestrzeganie zapewniało bezpieczeństwo określonych zasobów firmy (ludzi, sprzętu, danych i systemów) oraz właściwą obsługę nośników danych (zarządzanie nimi, niszczenie ich zawartości i ich samych). Musi definiować procedury postępowania z informacjami, wiedzą i dokumentacją systemową, sposób gromadzenia i udostępniania danych, umożliwiać ewentualny handel elektroniczny.

Zarówno zarządzanie wiedzą jak i polityka bezpieczeństwa są procesami ciągłymi, dlatego muszą podlegać monitorowaniu i modyfikowaniu. Przeobrażenia gospodarcze i technologiczne ku społeczeństwu informacyjnemu wymuszają traktowanie poruszanych zagadnień priorytetowo.

LITERATURA

- [Białas 2006] Białas A., *Bezpieczeństwo informacji i usług w nowoczesnej instytucji i firmie*, WNT, Warszawa 2006.
- [KPMG 1999] *Value Based Management: The growing importance of shareholder value in Europe*, A European Research Report, KPMG, 1999.
- [Mitnick 2003] Mitnick K., *Sztuka podstęp*, Helion, Gliwice 2003. Tytuł oryginału: *The Art of Deception: Controlling the Human Element of Security*.
- [PN 27001] PN-ISO/IEC 27001:2007 *Technika informatyczna. Techniki bezpieczeństwa. Systemy zarządzania bezpieczeństwem informacji. Wymagania*. PKN, Warszawa 2007.
- [Sandisk 2008] www.sandisk.com/Corporate/PressRoom/PressReleases/PressRelease.aspx?ID=4179 (9.04.2008).
- [Security 2008] www.securitystandard.pl/news/111965.html
- [Sveiby 1997] Sveiby E., *The New Organizational Wealth: Managing and Measuring Knowledge-Based Assets*. San Francisco, 1997.
- [www 2008] www.gocsi.com/

KNOWLEDGE MANAGEMENT AND SECURITY POLICY

More and more enterprises feel the need to introduce the ICT system and information security policy. A sudden development of information technologies, gathering and transfer of data requires additional actions connected with information protection and organized way of knowledge documentation and management. This paper covers issues related to knowledge management within the framework of the adopted security policy and the information security management system (ISMS).

*bezpieczeństwo danych,
bezpieczeństwo usług sieciowych,
ochrona komputera osobistego*

Hanna MAZUR*
Zygmunt MAZUR*

OCHRONA ZASOBÓW KOMPUTERA OSOBISTEGO

Posiadanie komputera osobistego jest dzisiaj standardem. Komputer jest zarówno narzędziem pracy, nauki, komunikacji i rozrywki, szczególnie w środowisku młodzieży uczącej się i studiującej. Standardem jest również dostęp do sieci Internet, korzystanie z jej zasobów i usług. Użytkownicy, świadomi korzyści jakie daje to narzędzie, często zapominają lub bagatelizują zagrożenia, jakie niesie ze sobą nieodpowiednie zabezpieczenie komputera i jego zasobów oraz nierozważne korzystanie z bogactwa Internetu. W artykule omówiono podstawowe zagrożenia dla bezpieczeństwa danych gromadzonych w komputerach osobistych i sposoby ich zabezpieczania. Przedstawiono również wyniki z ankiety sprawdzającej praktyczną realizację zabezpieczeń stosowanych przez ankietowaną grupę studentów.

1. WPROWADZENIE

Powszechnie znane jest powiedzenie, że jeśli chcemy, aby nasz komputer był bezpieczny, to nie należy podłączać go do sieci komputerowej i nie należy korzystać z żadnych nośników zewnętrznych (dyskietek, płyt CD, pendrive'ów) czyli trzeba go odizolować od świata zewnętrznego. Ale kto z nas tak pracuje? Tak jak standardem jest posiadanie komputera, tak i standardem jest, że korzystamy z dostępu do Internetu i wykorzystujemy zewnętrzne nośniki do przenoszenia danych między komputerami. A to oznacza, że zarówno nasz komputer jak i gromadzone dane już nigdy nie będą stuprocentowo bezpieczne. I musimy mieć świadomość, że zapewnienie możliwie jak najwyższego stopnia bezpieczeństwa jest procesem a nie jednorazową czynnością, którą na dodatek wykona za nas jakaś inna osoba (sprzedawca w sklepie ze sprzętem komputerowym lub serwisant).

O bezpieczeństwie naszego komputera decyduje wiele czynników, które omówimy w dalszej części artykułu. Niezwykle ważne w tym zakresie jest przestrzeganie kilku

* Instytut Informatyki, Wydział Informatyki i Zarządzania Politechniki Wrocławskiej, 50-370 Wrocław, Wybrzeże Wyspiańskiego 27, {hanna.mazur, zygmunt.mazur}@pwr.wroc.pl

zasad i dobrych praktyk. O ich realizację i stosowanie autorzy zapytali grupę ankietowanych. W rozdziale 4 przedstawiono wyniki z przeprowadzonego wśród studentów badania.

2. KONIECZOŚĆ OCHRONY KOMPUTERA OSOBISTEGO

Na początek zastanówmy się dlaczego, pomimo tylu znanych przykładów ataków komputerowych, nadal mają one miejsce? Dlaczego nie potrafimy zapewnić bezpieczeństwa fizycznego ani uodpornić się na stosowane metody socjotechniczne atakujących? Co sprawia, że oszuści komputerowi ciągle przeprowadzają udane ataki na internetowe konta bankowe? Czy robimy wszystko co moglibyśmy zrobić aby chronić swoje komputery, swoje dane i środki finansowe zgromadzone w banku?

Zanim spróbujemy znaleźć odpowiedź na te pytania sporządzmy listę powodów, dla którym musimy w ogóle o tym myśleć i przyczyn najpopularniejszych zagrożeń dla komputerów osobistych. Oczywiście podejmowane środki ochrony muszą być adekwatne do zasobów, które musimy zabezpieczać. Zapewne zabezpieczenia serwera bankowego muszą być inne niż komputera osobistego, ale utrata pliku z pracą magisterską może być tak samo (lub nawet bardziej dotkliwa) niż utrata pieniędzy. Prosty sposób na zabezpieczenie się przed utratą ważnego pliku jest sporządzenie jego kopii na nośniku zewnętrznym. Koszty ponoszone na zabezpieczanie komputera nie powinny przekraczać wartości zgromadzonych danych a zainstalowane oprogramowanie ochronne nie powinno powodować spadku wydajności komputera tym bardziej, że samo również może być źródłem zagrożenia.

Prezes firmy Kaspersky Lab. poinformował, że zaledwie w ciągu 2 dni w listopadzie 2008 r. zostało zaatakowanych 10 tysięcy serwerów w Europie i USA, między innymi metodą wstrzyknięcia kodu SQL (ang. *SQL Injection*). Ataki SQL Injection [WWWSQL 2008] są możliwe dzięki modyfikacji zapytania SQL kierowanego do bazy danych (np. przy wypełnianiu pól formularza na stronie internetowej). W wielu przypadkach, aby uzyskać dostęp do wybranego serwisu WWW należy podać nazwę użytkownika i hasło, tak jak to przedstawiono na rysunku 1. Gdyby do bazy danych przechowującej dane użytkowników (ich nazwy i hasła) zamiast poprawnego zapytania w języku PHP postaci

```
SELECT USER_ID FROM USERS WHERE (LOGIN='$user_login')
AND (PASS='$user_pass')
```

zostało skierowane zapytanie

```
SELECT USER_ID FROM USERS WHERE (LOGIN='admin') AND (PASS=''
OR '1'='1')
```


które powstanie, jeśli zamiast poprawnego hasła wstawiony będzie tekst (wstrzyknięty kod SQL)

```
' OR '1'='1
```

to atakujący zaloguje się jako administrator (jeśli wie, że administrator używa nazwy *admin*).

WYDZIAŁ INFORMATYKI I ZARZĄDZANIA

Witamy na oficjalnej stronie Wydziału Informatyki i Zarządzania!

LOGOWANIE

WYDZIAŁ STUDENCI PRACOWNICY DYDAKTYKA INFORMACJE REKRUTACJA SZUKAJ:

Strona główna » Logowanie do serwisu

Informatyka

- O kierunku
- Plany studiów i programy nauczania
- Katalog kursów
- Rozkłady zajęć
- Rekrutacja

Zarządzanie

- O kierunku
- Plany studiów i programy nauczania
- Rozkłady zajęć
- Rekrutacja
- Do strony www I23

Strona dla zarejestrowanych użytkowników

Zaloguj się

login :

hasło :

[zaloguj](#)

Logowanie

- [Przypomnij mi login i hasło](#)
- [Zgłoś problem administratorowi](#)

Dlaczego nie mogę się zalogować?

Użytkownikami uprawnionymi do logowania się w serwisie i posiadającymi konta w systemie są wszyscy pracownicy Wydziału Informatyki i Zarządzania, związani z dydaktyką.

Jeśli jesteś pracownikiem, a mimo to, nie możesz się zalogować, skorzystaj z [systemu przypominania haseł](#), lub [zgłoś problem administratorowi](#).

Jeśli nie jesteś uprawnionym pracownikiem, Niestety ta część serwisu jest dla Ciebie niedostępna.

- [Przypomnij mi login i hasło](#)
- [Zgłoś problem administratorowi](#)

SAMORZĄD STUDENCKI

- PRACJA
- PROJEKTY
- STIPREZY

Copyright © 2012-2014 by Wydział Informatyki i Zarządzania PWr, created by Gryx & Falwaki

Rys. 1. Przykładowy formularz umożliwiający logowanie się do serwisu WWW

Główne zagrożenia dla komputera osobistego, wynikające z korzystania z usług internetowych, to między innymi:

- zainfekowanie wirusami i robakami komputerowymi, których działanie może być mniej lub bardziej dokuczliwe lub niszczące,
- zainstalowanie koni trojańskich,
- zainstalowanie oprogramowania szpiegującego (ang. *spyware*) pracującego w tle innych aplikacji,
- spam – otrzymywanie niechcianej korespondencji drogą elektroniczną,
- podsłuchiwanie (ang. *sniffing*), przechwytywanie i analizowanie, przez nieupoważnione osoby, informacji do nas przesyłanych i odbieranych,

- przejęcie przez atakującego nazw użytkowników i haseł,
- *phishing* (pozyskiwanie haseł) – kradzież i oszukiwanie ludzi poprzez np. tworzenie fikcyjnych stron WWW identycznych ze stronami banków, sklepów lub aukcji.
- zainstalowanie oprogramowania monitorującego pracę komputera (ang. *keylogger*) czyli niewidocznego programu przechwytyującego i przekazującego do zdalnego komputera informacji związanych z wszelką aktywnością użytkownika np. prywatną korespondencję, wpisywane znaki z klawiatury, odwiedzane strony, treści z komunikatorów internetowych,
- nieupoważniony dostęp do danych, utrata danych,
- przejęcie kontroli nad komputerem przez atakującego,
- atak przez otwarte porty (np. w przypadku zainstalowanego serwera FTP czy WWW, komunikatora lub oprogramowania P2P),
- farming (ang. *pharming*) czyli zmiana adresów DNS (ang. *Domain Name System*), przez co użytkownik nieświadomie wchodzi na fałszywą stronę WWW wykorzystywaną do przechwycenia poufnych danych, np. o jego karcie kredytowej.

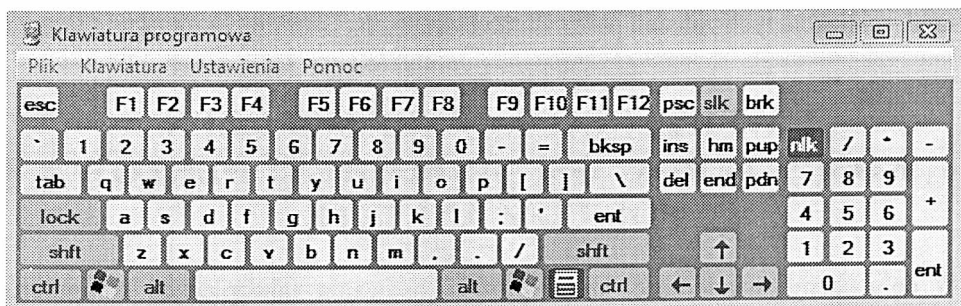
Aby zabezpieczyć komputer przed zagrożeniami, których przykłady wymieniono, należy stosować się do zaleceń, dotyczących bezpieczeństwa komputerowego, przedstawionych w następnym rozdziale.

3. SPOSOBY OCHRONY KOMPUTERA OSOBISTEGO

Ponieważ najwięcej zagrożeń wynika z podłączenia komputera do sieci i korzystania z usług sieciowych oraz przenoszenia danych między komputerami za pomocą zewnętrznych nośników danych, stąd stosowane zabezpieczenia muszą przede wszystkim dotyczyć właśnie tych obszarów. Podstawowe sposoby zabezpieczeń, możliwe do stosowania przez nawet początkujących użytkowników i bez dużych nakładów finansowych, to:

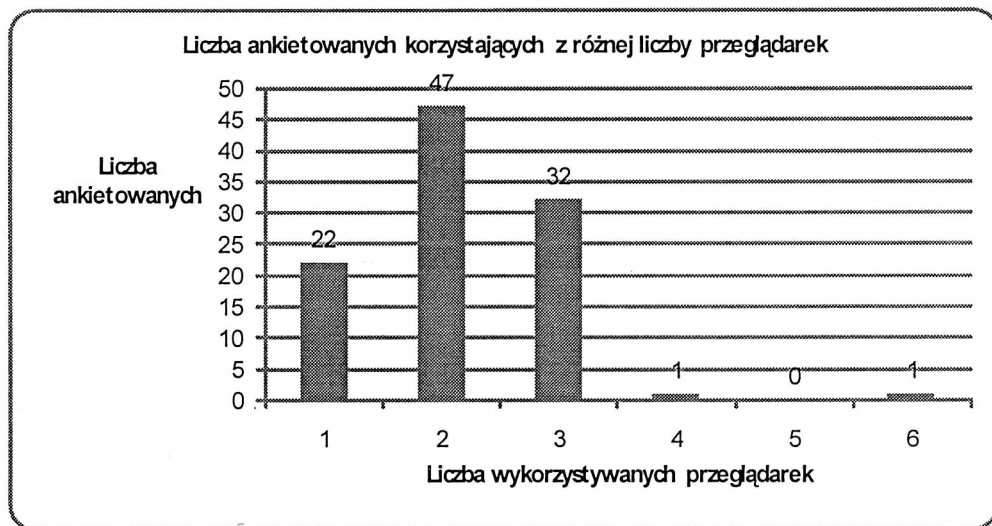
- aktualizacja oprogramowania ochronnego i bazy wirusów,
- skanowanie komputera ze względu na obecność programów szpiegujących i usuwanie ich z komputera,
- odpowiednie stosowanie, modyfikowanie i przechowywanie haseł dostępu,
- usuwanie historii przeglądanych stron i plików tymczasowych,
- sprawdzanie, czy są na komputerze otwarte porty,
- ograniczanie korzystania z kont na portalach społecznościowych lub odpowiednie ich konfigurowanie,
- kontrolowanie informacji pozostawianych na stronach WWW,
- nie korzystanie z kont osobistych na komputerach ogólnodostępnych (ewentualnie częste zmiany haseł),

- trwale kasowanie plików (za pomocą klawiszy *Shift +Delete*), systematyczne usuwanie plików z folderu *Kosz*,
- kontrolowanie zabezpieczeń komputera, ich aktualizowanie i dostosowywanie do aktualnych potrzeb i zmian,
- automatyczne aktualizowanie systemu operacyjnego,
- ochrona bufora przed przepełnieniem – w wyniku błędu w kodzie programu możliwe jest wczytanie do bufora (wyznaczonego obszaru pamięci), zbyt dużej ilości danych. Service Pack 2 dla systemu Windows XP zawiera program DEP (ang. *Data Execution Prevention*), którego celem jest ochrona przed tego typu przepełnieniem stosu i wywołaniem spreparowanego kodu. Niestety i te zabezpieczenia można ominąć,
- fizyczne zabezpieczanie komputera; komputer, tak jak każdy inny sprzęt, ma określony czas życia i niestety też ulega zużyciu a uszkodzenia fizyczne są dużo poważniejsze od uszkodzeń logicznych,
- tworzenie kopii zapasowych ważnych danych, a w szczególności danych kontaktowych, książek adresowych, plików o istotnym znaczeniu, ustawień konfiguracyjnych itd., na wypadek awarii komputera lub utraty komputera (wskutek kradzieży lub zgubienia),
- przechowywanie ważnych plików jako ukrytych lub w folderach ukrytych,
- ukrywanie adresu IP komputera np. poprzez wykorzystanie serwera pośredniczącego (ang. *proxy*); stosowanie serwera proxy zmniejsza ruch sieciowy na dużych odległościach i przyspiesza otwieranie stron WWW, ponadto serwer proxy jest dodatkowym elementem zabezpieczającym, ponieważ w warstwie aplikacyjnej pracuje w modelu ISO/OSI a to oznacza, że może on sprawdzać nie tylko formalną poprawność przesyłanych pakietów ale także logiczną zawartość,
- usuwanie plików *cookies* (ciasteczek) lub automatyczne umieszczanie ich w folderze *Kosz*,
- wykorzystywanie klawiatury programowej (rys. 2) do wprowadzania haseł w celu zabezpieczenia przed przechwytywaniem informacji o klawiszach naciśniętych na klawiaturze,



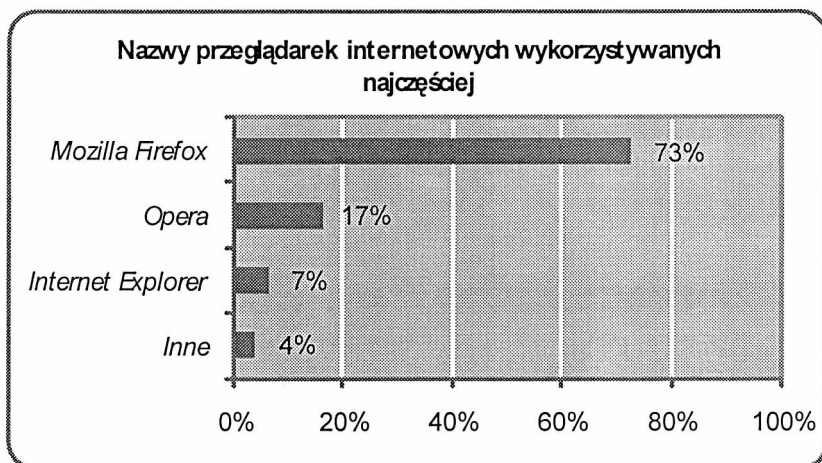
Rys. 2. Klawiatura programowa do bezklawiaturowego wprowadzania znaków, np. haseł

Studenci na ogół wykorzystują różne przeglądarki internetowe – 81 osób (79%) korzysta z co najmniej dwóch przeglądarek. Maksymalna liczba wykorzystywanych przeglądarek podanych w ankiecie to sześć (rys. 4).



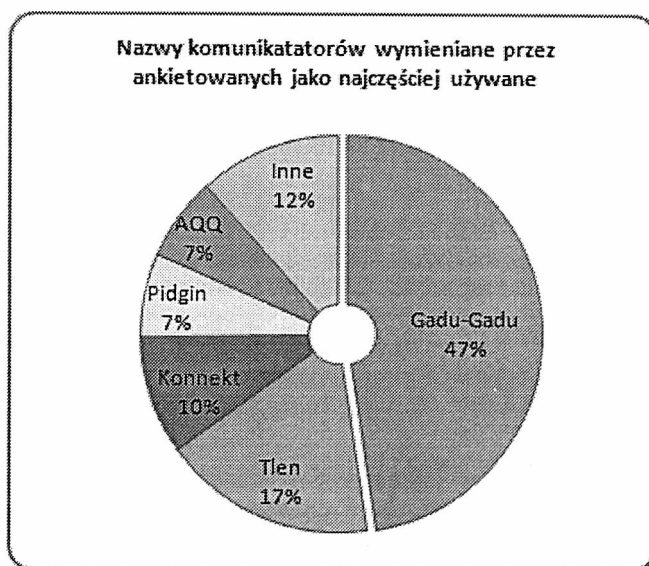
Rys. 4. Liczbowe zestawienie wykorzystywania przeglądarek internetowych

Spośród przeglądarek internetowych najpopularniejszą jest Mozilla Firefox, którą wykorzystuje 73% ankietowanych (rys. 5). Przeglądarka ta oferuje przyjazny interfejs, szybką i wydajną pracę, ma wbudowane mechanizmy bezpieczeństwa.

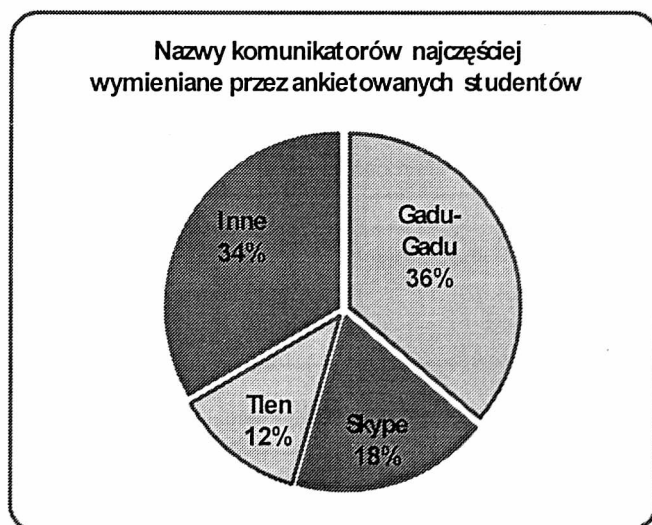


Rys. 5. Nazwy przeglądarek wykorzystywanych najczęściej przez ankietowanych

Największą popularnością spośród komunikatorów internetowych cieszy się wśród ankietowanych Gadu-Gadu, który był wymieniany przez ankietowanych zarówno najczęściej wymieniany (nie zawsze jako pierwszy) oraz jako najczęściej używany (rys. 6 i 7).



Rys. 6. Nazwy komunikatorów wymieniane przez ankietowanych jako pierwsze



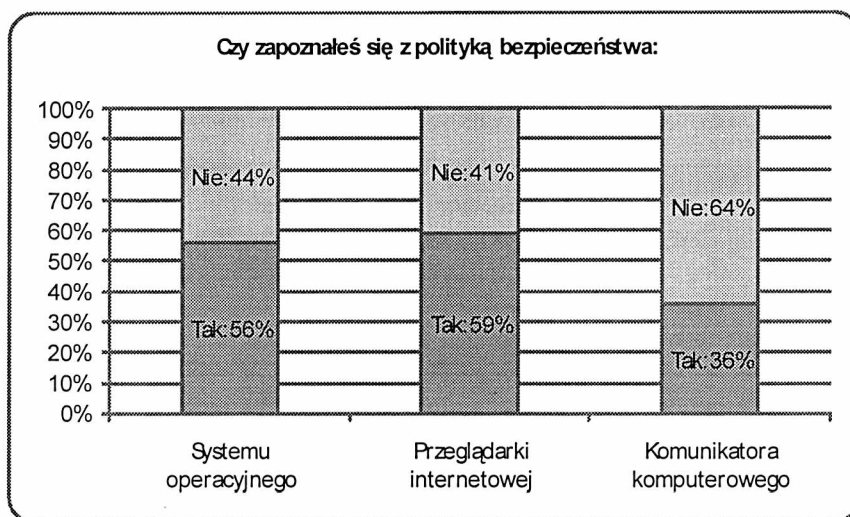
Rys. 7. Nazwy komunikatorów najczęściej wykorzystywanych przez ankietowanych

Otrzymane wyniki ankiety potwierdzają wysoką pozycję komunikatora Gadu-Gadu w Polsce, który jest liderem w tym obszarze zastosowań. Wpływ na jego popularność ma zapewne możliwość korzystania z niego bez konieczności instalacji na komputerze. Wystarczy wejść na stronę <http://web.gadu-gadu.pl/>, która umożliwia korzystanie z komunikatora za pomocą przeglądarki internetowej.

Kolejne pytanie składało się z trzech podpunktów, które miały na celu sprawdzenie, czy ankietowani zapoznali się z polityką bezpieczeństwa wykorzystywanego systemu operacyjnego, przeglądarki internetowej i komunikatora internetowego. Liczbowe zestawienie wyników odpowiedzi na to pytanie przedstawiono w tabeli 1 a procentowe zilustrowano na rysunku 8.

Tabela 1. Liczbowe zestawienie odpowiedzi ankietowanych
odnośnie zapoznania się z polityką bezpieczeństwa

Czy zapoznałeś się z polityką bezpieczeństwa	Tak	Nie	Razem
Systemu operacyjnego	58	45	103
Przeglądarki internetowej	61	42	103
Komunikatora komputerowego	37	66	103



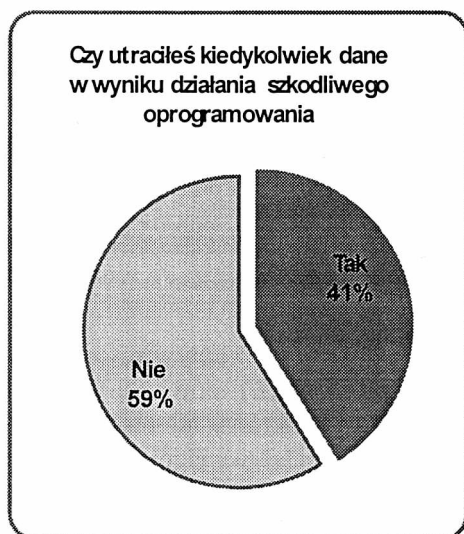
Rys. 8. Procentowe zestawienie odpowiedzi ankietowanych
dotyczących zapoznania się z polityką bezpieczeństwa systemów operacyjnych,
przeglądarki internetowej i komunikatora komputerowego

Analiza wyników ankiety wykazuje, że najmniej uwagi (37%) ankietowani przywiązują do polityki prywatności komunikatorów komputerowych (typu peer-to-peer), która jest łatwo dostępna w Internecie (na przykład polityka prywatności i informacje

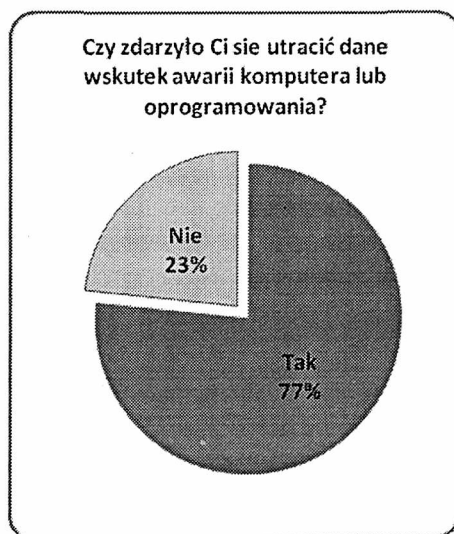
o bezpieczeństwie w Skype są umieszczone na stronie www.skype.com). Tak małe zainteresowanie ogłaszanymi politykami bezpieczeństwa zapewne wynikają z założenia, że niezależnie od przyjętej polityki i tak będziemy z tego sposobu komunikacji korzystali. Ponadto, wiele osób ma ograniczone zaufanie do ogłaszanej polityki prywatności, która nie zawsze jest przez firmy respektowana. Na przykład z raportu *California HealthCare Foundation* wynika, że wiele witryn internetowych zajmujących się problematyką ochrony zdrowia łamie własną politykę ochrony prywatności. Utrata prywatności, spowodowana rozwojem nowoczesnych technologii, jest problemem obecnych czasów.

4.2. WERYFIKACJA ZALECEŃ ODNOŚNIE BEZPIECZEŃSTWA

Brak zabezpieczeń komputera ułatwia przeprowadzanie na niego ataków. Zaatakowany komputer nie tylko udostępnia swoje dane ale jednocześnie może być wykorzystywany do rozsyłania spamu lub przeprowadzania kolejnych ataków (bez wiedzy jego właściciela). Komputery takie nazywane są zombi. Z wrześniowego raportu firmy Symantec wynika, że stale rośnie liczba komputerów zombi oraz ilość rozsyłanego spamu zawierającego złośliwe oprogramowanie. Polska pod względem liczby komputerów zombi zajmuje, niestety, wysokie czwarte miejsce – 4% komputerów zombi znajduje się w Polsce (12% w Turcji, 9% w Brazyli i 8% w Rosji).



Rys. 9. Utrata danych wskutek szkodliwego oprogramowania



Rys. 10. Utrata danych wskutek awarii komputera lub oprogramowania

W kontekście publikowanych raportów przedstawiających stale rosnące zagrożenia bezpieczeństwa komputerowego pochodzące z Internetu, zasadne wydają się kolejne pytania w ankiecie.

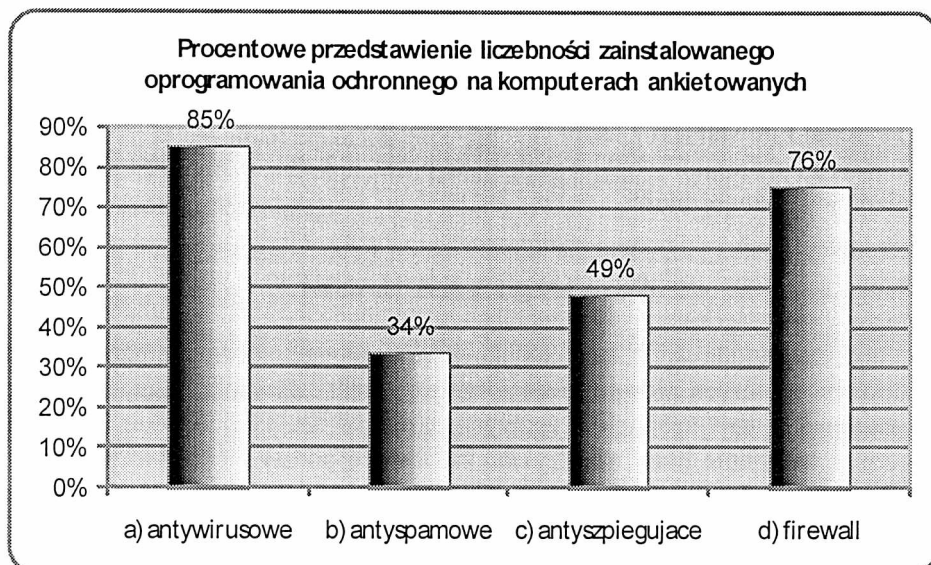
Zadaniem ankietowanych było udzielenie odpowiedzi TAK lub NIE na kolejnych pięć pytań. Z udzielonych odpowiedzi wynika, że aż 41% ankietowanych utraciło dane w skutek działania szkodliwego oprogramowania (rys. 9) i aż 77% – w wyniku awarii komputera lub oprogramowania (rys. 10).

O tym, jak małą wagę przywiązuje się do bezpieczeństwa w Internecie świadczy fakt, że aż 74% ankietowanych ignoruje komunikat, że strona WWW nie jest bezpieczna i nie rezygnuje z jej przeglądania (rys. 11). Sytuacja taka jest niebezpieczna ponieważ użytkownik, który wielokrotnie korzysta ze stron bezpiecznych, poprzedzonych takim komunikatem, przyzwyczaja się do niego i właściwie już go nie czyta i na niego nie reaguje, a to z kolei znacznie obniża jego czujność.

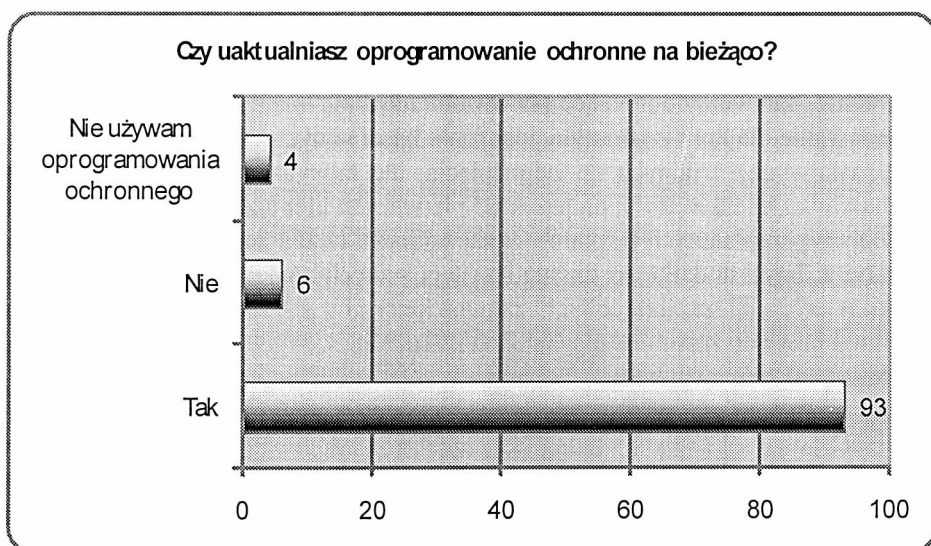


Rys. 11. Wykres reakcji ankietowanych na komunikat o niebezpiecznej stronie WWW

O konieczności instalowania oprogramowania ochronnego – antywirusowego, antyspamowego, antyszpiegującego i zaporowego (ang. *firewall*) mówi się i pisze tak dużo, że trochę dziwi niefrasobliwość 15% ankietowanych, którzy nie mają zainstalowanego oprogramowania antywirusowego i 24% bez zainstalowanego oprogramowania zaporowego. Procentowe przedstawienie liczebności zainstalowanego oprogramowania na komputerach ankietowanych przedstawiono w formie wykresu na rysunku 12. Pozytywny wynik uzyskano w pytaniu o aktualizowanie oprogramowania ochronnego – 93 osoby (co stanowi 90% ankietowanych) uaktualnia je na bieżąco, co pokazano na rysunku 13.



Rys. 12. Procentowa ilustracja zainstalowanego oprogramowania ochronnego przez ankietowanych



Rys. 13. Liczba osób aktualizujących oprogramowanie ochronne

Ankietowani najczęściej wymieniali następujące nazwy oprogramowania zainstalowanego jako:

- oprogramowanie antywirusowe Avast Antivirus (27 osób), AVG (14 osób),

- oprogramowanie antyspamowe Kaspersky (5 osób),
- oprogramowanie antyszpiegujące Ad-Adware (12 osób),
- zaporę systemową Windows (13 osób), zapory COMODO, Kaspersky, Sygate Personal Firewall (po 7 osób).

Pozostałe inne programy były wymieniane przez 1–3 osób a więc jest w tym zakresie bardzo duża różnorodność.

4.3. POLITYKA HASEŁ

Pomimo, że metoda weryfikacji użytkowników poprzez hasła obarczona jest wieloma wadami, z których główne to konieczność przechowywania hasel, zapominanie hasel, stosowanie zbyt krótkich i typowych hasel, przekazywanie hasel innym użytkownikom, przesyłanie hasel na przykład za pomocą poczty elektronicznej lub SMS (ang. *Short Message Service* – usługa przesyłania krótkich wiadomości tekstowych, które mogą być wysyłane z telefonów komórkowych lub ze stron WWW), to jest ona bardzo często stosowana. Stąd tak duże znaczenie ma znajomość podstawowych zaleceń odnośnie stosowania hasel i ich przestrzeganie.

Używanie i zarządzanie hasłami reguluje norma PN-ISO/IEC 27001 [Norma 2007], punkt A.11. „Kontrola dostępu”, a w szczególności:

- punkt A.11.3.1: *Używanie hasel*,
- punkt A.11.5.3: *System zarządzania hasłami*.

Najważniejsze zasady dotyczące stosowania hasel to:

- nadawanie dla kont o istotnym znaczeniu hasel silnych czyli długich (minimalnie 8-znakowych), trudnych do odgadnięcia, ale łatwych do zapamiętania ciągów znaków całej klawiatury, na przykład pierwszych liter kolejnych słów ulubionego wiersza przeplatanych liczbami (oznaczającymi liczbę znaków w słowie lub liczbami z daty), dodatkowo czasami zastępowanych znakami specjalnymi („\$” zamiast „s”, „@” zamiast „a”, „!” zamiast litery „l”, „1” zamiast „i”), takie hasła są silne i łatwe do zapamiętania (np. \$n\$!c01pzn\$),
- stosowanie różnych hasel do różnych kont,
- zapamiętywanie hasel – hasel nie należy nigdzie zapisywać, można ewentualnie przechowywać zaszyfrowane lub przechowywać w bezpiecznym, trudno dostępnym, dla osób nieuprawnionych, miejscu (np. dla bezpieczeństwa systemów informatycznych administratorzy, na wszelki wypadek, muszą hasła zapisywać i bezpiecznie przechowywać),
- nie ujawnianie hasel innym osobom,
- okresowa zmiana hasel,
- modyfikowanie hasel po zaobserwowaniu sytuacji nietypowych,
- wprowadzanie czasami błędnych hasel podczas logowania (jeśli zostanie przechwycone, to uniemożliwi przejęcie kontroli nad kontem).

Liczba możliwych haseł o długości n znaków zapisanych za pomocą alfabetu o m znakach wynosi m^n . Podstawowe metody łamania haseł to:

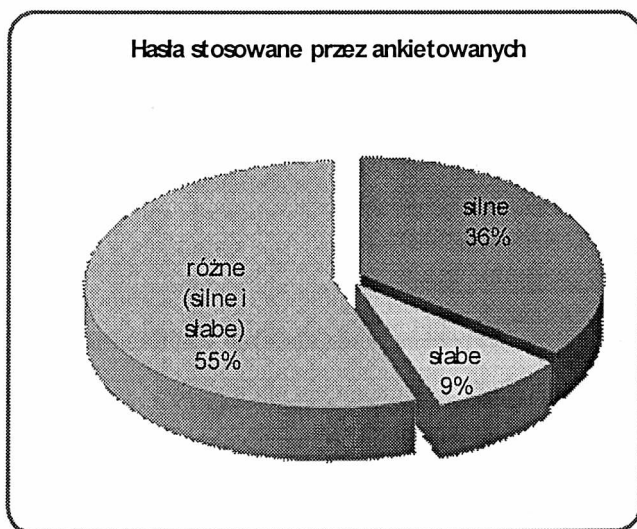
- metoda siłowa czyli brutalny atak (ang. *brute force*), polegająca na generowaniu i sprawdzaniu kolejnych haseł,
- metoda słownikowa (ang. *word list*), polegająca na wykorzystywaniu łańcuchów ze słowników haseł.

Zgodnie z Rozporządzeniem Ministra Spraw Wewnętrznych i Administracji z dnia 29 kwietnia 2004 r. do Ustawy z 29 sierpnia 1997 r. *O ochronie danych osobowych*, w którym zostały określone trzy poziomy bezpieczeństwa przetwarzania danych osobowych w systemach informatycznych (podstawowy, podwyższony, wysoki), zdefiniowano odpowiednio trzy poziomy dla polityki haseł:

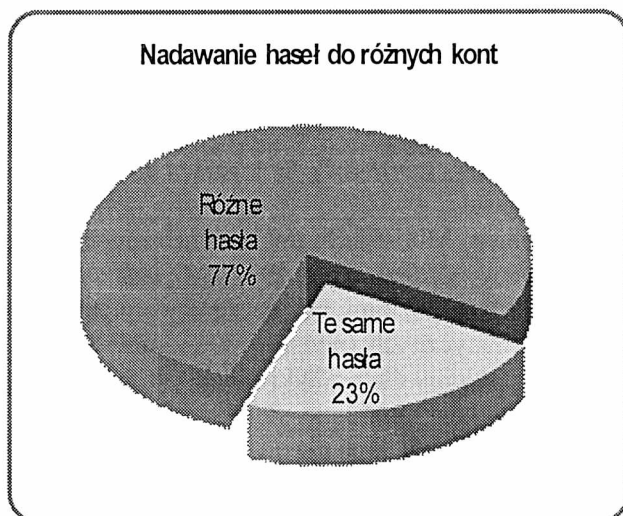
- poziom podstawowy – hasło powinno się składać minimum z 6 znaków oraz powinno być zmieniane nie rzadziej niż co 30 dni,
- poziom podwyższony oraz poziom wysoki – hasło powinno się składać minimum z 8 znaków, zawierać małe i wielkie litery oraz cyfry lub znaki specjalne, i powinno być zmieniane nie rzadziej niż co 30 dni.

Przeprowadzona wśród studentów ankieta wykazała, że 36% ankietowanych stosuje hasła silne a 55% słabe i silne (rys. 14), ponadto aż 77% ankietowanych nadaje różne hasła do różnych kont (rys. 15).

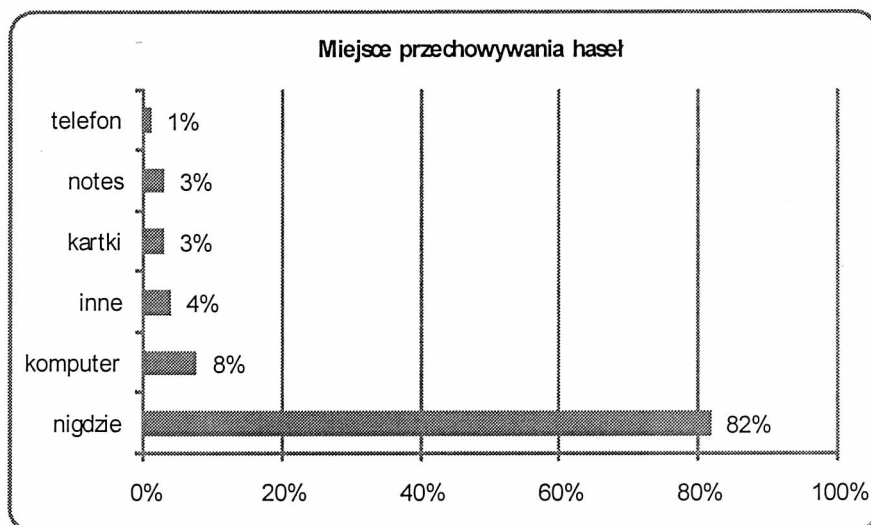
Wyniki badań wykazują, że pracownicy beztrudno zapisują hasła na kartkach, które dla wygody umieszczają w widocznym miejscu przy komputerze. Studenci znacznie lepiej wypadli pod tym względem. Aż 82% ankietowanych nie zapisuje nigdzie haseł (rys. 16).



Rys. 14. Procentowe zestawienie stosowania haseł silnych i słabych

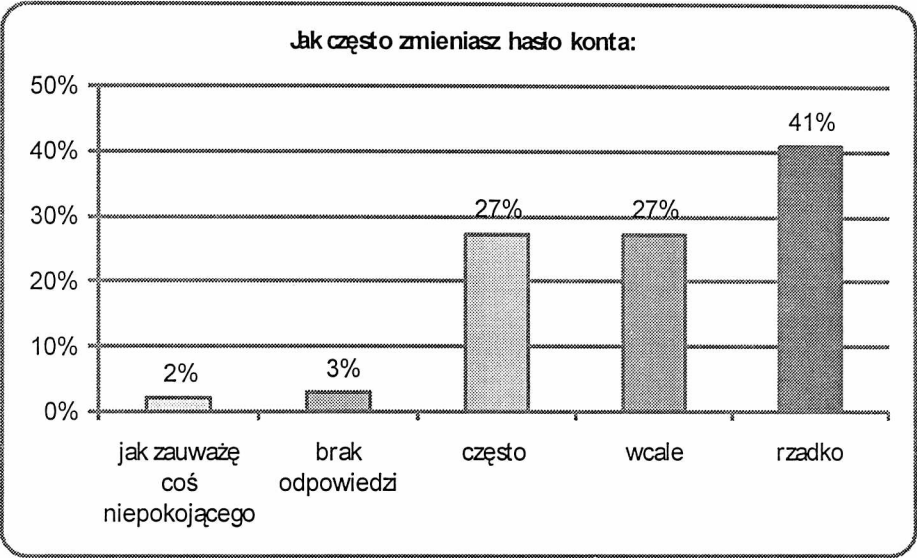


Rys. 15. Procentowe zestawienie stosowania różnych haseł do różnych kont



Rys. 16. Miejsca przechowywania haseł do kont internetowych

Z analizy odpowiedzi wynika, że 27% ankietowanych często zmienia hasła a natomiast rzadko lub wcale aż 68%, 2% – gdy zauważą coś niepokojącego a 3% nie udzieliło odpowiedzi na to pytanie (rys. 17). Odpowiedzi te są istotne w kontekście odpowiedzi na następne pytanie. Ponad 83% ankietowanych przeprowadza transakcje finansowe przez Internet (rys. 18) przy czym 18% spośród nich określiło, że są to transakcje powyżej 5 000 zł rocznie (rys. 19).

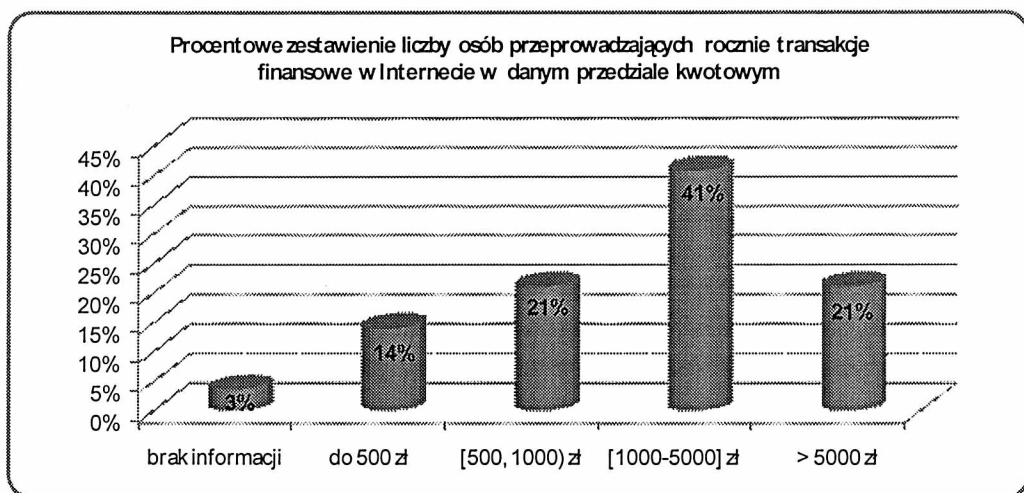


Rys. 17. Częstości zmiany haseł do kont internetowych

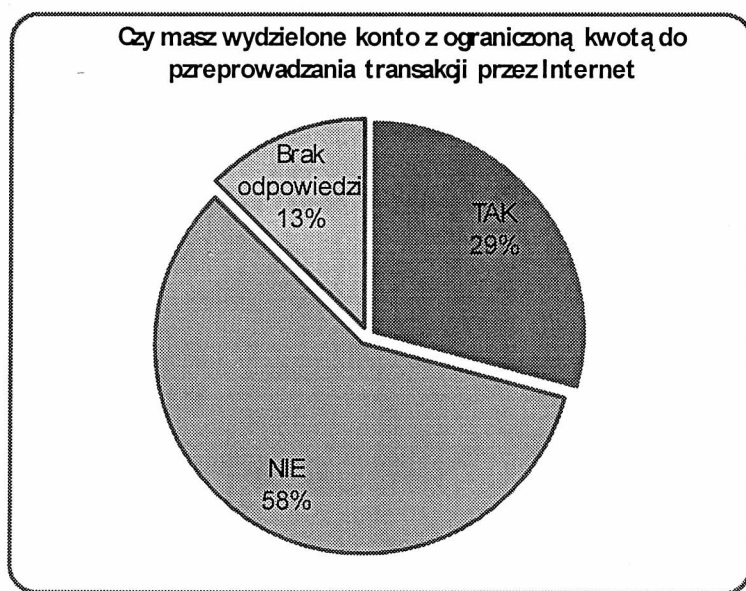


Rys. 18. Ilustracja przeprowadzania transakcji przez Internet

W celu zwiększenia bezpieczeństwa finansowego dobrze jest założyć konto z niewielką kwotą. Z ankiety wynika, że tylko 29% ankietowanych ma wydzielone konto z ograniczoną kwotą pieniędzy do przeprowadzania transakcji internetowych (rys. 20).



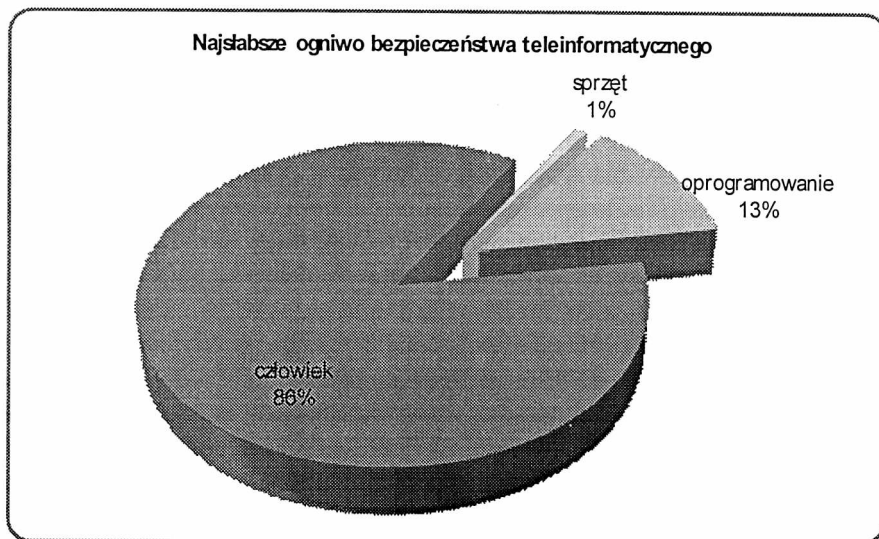
Rys. 19. Roczne przedziały kwotowe transakcji internetowych ankietowanych



Rys. 20. Ilustracja posiadania odrębnych kont z ograniczonymi środkami pieniężnymi do przeprowadzania finansowych transakcji przez Internet

W ostatnim pytaniu zadaniem ankietowanych było wskazanie najsłabszego ogniw bezpieczeństwa teleinformatycznego spośród wymienionych: sprzęt, oprogramowanie, człowiek. Można było zaznaczyć dowolną liczbę odpowiedzi. Z rysunku 21 wynika, iż ankietowani mają świadomość, że najsłabszym ogniwem bezpieczeństwa teleinfor-

matycznego jest człowiek czyli, że najłatwiej dostać się do zabezpieczonych danych przez człowieka – na przykład pracownika świadomie lub nieświadomie nieprzestrzegającego obowiązujące procedury bezpieczeństwa.



Rys. 21. Najslabsze ogniwo bezpieczeństwa teleinformatycznego w ocenie ankietowanych

5. PODSUMOWANIE

Znajomość zagrożeń pochodzących z sieci komputerowej, znajomość sposobów zabezpieczenia się przed nimi i stosowanie tych zabezpieczeń pozwala na w miarę bezpieczne gromadzenie danych na komputerze osobistym i korzystanie z usług sieciowych. Przestrzeganie wymogów bezpieczeństwa teleinformatycznego w celu ochrony informacji jest przedmiotem licznych badań i ankiet, których wyniki pozwalają sprawdzić i określić poziom świadomości kadry kierowniczej, osób odpowiedzialnych za realizację przyjętej polityki bezpieczeństwa w organizacji oraz pozostałych pracowników w zakresie opracowywania i stosowania odpowiednich procedur, systematyczności wykonywania odpowiednich zadań i odpowiedzialności poszczególnych osób w tym obszarze. Prawie każdy użytkownik sieci doświadczył w praktyce zagrożenia bezpieczeństwa informatycznego chociażby w postaci niechcianego spamu czy wykrycia wirusa na swoim komputerze. Z badania przeprowadzonego w 2006 roku przez Instytut Millward/Brown SMG/KRC wynika, że ponad 33% firm i prawie 50% użytkowników indywidualnych poniosło straty wskutek ataków skierowanych przeciwko systemom komputerowym przy czym 85% badanych jako jedyny sposób zabezpieczenia komputera wymieniło zainstalowanie oprogramowania antywirusowego.

Konsekwencje za nieprzestrzeganie obowiązujących przepisów są ściśle określone – i tak na przykład administrator bazy danych, który nie przestrzega wymogów obowiązującej ustawy o ochronie danych osobowych i rozporządzeń wykonawczych (dotyczących rejestrowania bazy danych osobowych, legalności przetwarzania informacji oraz stosowanych zabezpieczeń organizacyjnych i technicznych), jest zagrożony karą dyscyplinarną, administracyjną (nakazującą usunięcie uchybień i naprawienie błędów), finansową (karą grzywny) lub nawet karą pozbawienia wolności do lat 3 [Ustawa 1997].

W rozdziale przedstawiono konieczność i podstawowe sposoby ochrony komputera osobistego i zminimalizowania ryzyka utraty danych poprzez systematyczne stosowanie się do proponowanych zaleceń. Następnie przedstawiono wyniki autorskiej ankiety dotyczącej bezpieczeństwa komputerowego przeprowadzonej wśród studentów III roku informatyki uczelni wyższej. Celem ankiety było poznanie stopnia świadomości zagrożeń oraz sprawdzenie i ocena możliwości i stosowanych przez studentów sposobów zabezpieczeń w zakresie bezpieczeństwa informacji. Ankietowana grupa osób, z racji wybranego kierunku studiów, wykorzystuje komputery do codziennej pracy i nauki, posiada co najmniej podstawową znajomość zagadnień z technologii informacyjnych, korzysta na co dzień z usług sieci Internet, potrafi wyszukać odpowiednie bezpłatne programy ochronne w sieci i zainstalować je na swoim komputerze, zna języki obce. Wyniki ankiety wykazały, że ankietowani niejednokrotnie doświadczyli utraty danych i awarii komputera, przeprowadzają transakcje finansowe przez Internet a jednak nie wszyscy przestrzegają podstawowe zasady bezpieczeństwa (choćby na przykład poprzez instalowanie oprogramowania ochronnego).

Interesujące byłoby porównanie otrzymanych wyników ankiety z wynikami ankiety przeprowadzonej wśród studentów innej uczelni i kierunku na przykład kierunku humanistycznego. Autorzy planują przeprowadzić taką ankietę w najbliższym czasie.

LITERATURA

- [Norma 2007] PN-ISO/IEC 27001:2007 *Technika informatyczna. Techniki bezpieczeństwa. Systemy zarządzania bezpieczeństwem informacji. Wymagania*. PKN, Warszawa 2007.
- [Ustawa 1997] *Ustawa z 29 sierpnia 1997 roku o ochronie danych osobowych*. <http://isip.sejm.gov.pl>.
- [WWWSQL 2008] http://www.poradnik-webmastera.com/artykuly/bazy_danych/sql_injection.php.

PROTECTION OF PC RESOURCES

Possessing a PC is a standard today. Computer is an instrument of work as well as entertainment, particularly within the teenager and student environments. Internet access and using its resources and services is also a standard. Conscious of the benefits that this tool gives us, we often forget or underestimate the threats that come with inappropriate security of computers and careless use of Internet's resources. The paper discusses basic threats to the security of data stored in PCs and the methods of their protection. It also presents a realization of protection used in practice by a questioned group of students.

Teresa MENDYK-KRAJEWSKA*

ZAGROŻENIA SYSTEMÓW SIECI ROZLEGŁEJ – SKALA ZJAWISKA

Praca dotyczy zagrożenia bezpieczeństwa danych systemów sieci rozległej. Wobec niebывале-go wzrostu zagrożeń sieciowych problem ten nabiera coraz większego znaczenia. Problematyką tą zajmują się wyspecjalizowane instytucje, które systematycznie publikują wyniki swoich badań. Obecnie rzadziej występują zwykłe wirusy czy robaki, na czele rejestrowanych zagrożeń pojawiły się programy szpiegujące i konie trojańskie. Przestępcy sieciowi, najczęściej atakujący serwery i aplikacje internetowe, działają w zorganizowanych grupach, a ich celem jest uzyskanie korzyści finansowych.

Do przeprowadzenia ataku często wykorzystywane są wady oprogramowania. Producenci systemów możliwie szybko usuwają zidentyfikowane luki, jednak nieustannie odkrywa się kolejne. Nowy rodzaj zagrożenia stanowią dystrybucja zainfekowanych nakładek systemowych oraz sprzedaż fałszywego oprogramowania ochronnego w celu uzyskania korzyści finansowych. Problem jest niebagatelny z uwagi na skalę zjawiska.

1. WSTĘP

Sieci rozległe i systemy z nimi połączone nie były i nie są bezpieczne, a obserwowany w ostatnim okresie wzrost zagrożeń danych i usług sieciowych może budzić uzasadniony niepokój. Zagrożenie może wynikać z zastosowania domyślnej nazwy, zbyt słabego hasła, z błędów oprogramowania lub nieprawidłowej konfiguracji systemu. Celem ataku może być każdy serwer sieciowy i każda aplikacja internetowa udostępniana on-line. Atakowane są przede wszystkim portale o znaczeniu strategicznym, ale też fora dyskusyjne, galerie zdjęć, serwisy społecznościowe oraz prywatne strony internetowe. Powodów prowadzenia przestępczej działalności w sieci jest wiele, a przede wszystkim intratność takich działań przy możliwości zachowania anonimowości. Główny problem, jak się zdaje, stanowią przyczyny, dla których można skutecznie prowadzić takie destrukcyjne, niezgodne z prawem dzia-

* Instytut Informatyki, Wydział Informatyki i Zarządzania Politechniki Wrocławskiej, 50-370 Wrocław, Wybrzeże Wyspiańskiego 27, teresa.mendyk-krajewska@pwr.wroc.pl

łania, a zatem istnienie luk w oprogramowaniu systemowym i użytkowym, samych aplikacjach ochronnych oraz stosowanych protokołach, które pozwalają na bezprawne wtargnięcie do systemu szkodliwym kodom o różnorodnym przeznaczeniu. Znawcy przedmiotu podkreślają, iż na przestrzeni ostatnich kilku lat natura zagrożeń bezpieczeństwa w sieci uległa zmianie, i tak „sportowa” działalność destrukcyjna ustąpiła profesjonalnej aktywności przestępczej. Włamywacze sieciowi zdeterminowani możliwością osiągnięcia ogromnych korzyści finansowych stają się coraz bardziej pomysłowi i bezwzględni.

Utrzymanie osiągniętego poziomu zabezpieczeń systemu nie jest łatwym zadaniem z uwagi na szybkość pojawiania się nowych zagrożeń, skuteczność maskowania szkodliwych kodów, niedostateczną wiedzę użytkowników z zakresu użytkowania i zabezpieczania systemu oraz złożoność i mnogość czynności podejmowanych w celu osiągnięcia możliwie maksymalnego poziomu ochrony systemu.

2. SKALA ZAGROŻENIA SIECI INTERNET

Niestety, zjawisko zagrożenia bezpieczeństwa sieciowego nieustannie rośnie. Aktywność przestępców internetowych postępuje bowiem w ślad za nieustannie zwiększającą się popularnością Internetu, w tym usług sieciowych. Oczywiście jest, że sposoby skutecznego atakowania systemów oraz maskowania obecności szkodliwych kodów rozwijają się wraz z rozwojem technik zabezpieczania systemów.

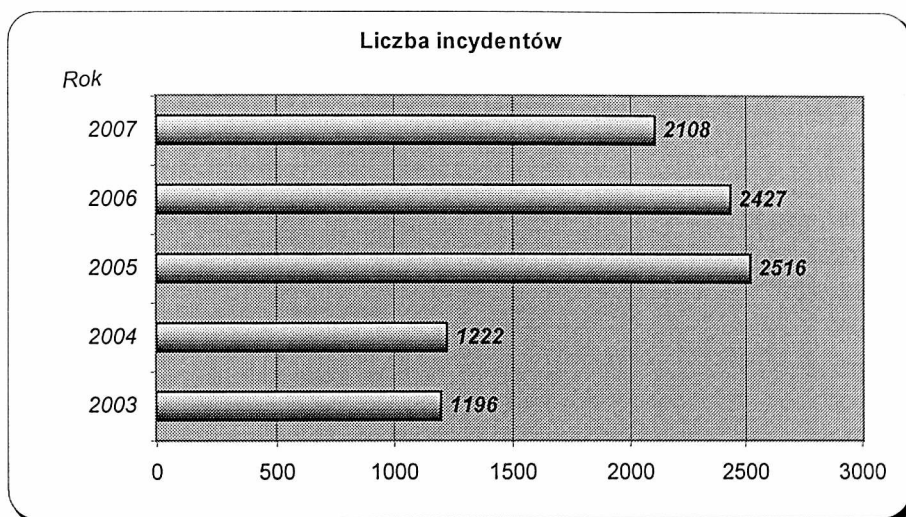
Z raportu z 2007 roku firmy Symantec¹ wynika, że liczba wszystkich znanych wirusów, robaków, koni trojańskich² i tym podobnych zagrożeń przekroczyła już jeden milion typów. Większość z nich wykorzystuje luki systemu operacyjnego MS Windows, będącego najpopularniejszą platformą systemową [Sym].

Instytucje zajmujące się zjawiskiem zagrożenia bezpieczeństwa sieci globalnej systematycznie publikują efekty swoich badań. Z danych Raportu 2007 zespołu CERT Polska³ widać, iż w ciągu ostatnich dwóch lat zmniejszyła się liczba rejestrowanych incydentów (patrz rysunek 1). Ten niewielki spadek tłumaczy się jednak niewuwzględnianiem informacji pochodzących z systemów automatycznych (są one bezpośrednio kierowane do właściwych operatorów) oraz niewystępowaniem w tym okresie epidemii robaków sieciowych. Rozkład najczęściej występujących podtypów zarejestrowanych incydentów przedstawiono na rysunku 2.

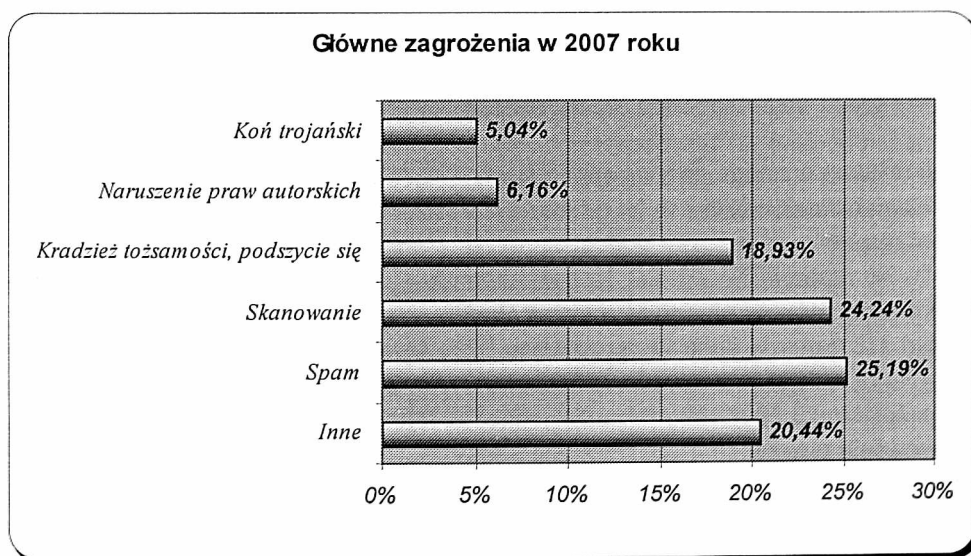
¹ Jedna z wiodących firm na rynku oprogramowania ochronnego, publikująca na koniec każdego roku wyniki swoich badań.

² Programy podszywające się pod inne aplikacje, a umożliwiające przejęcie kontroli nad systemem.

³ *Computer Emergency Response Team Polska* – zespół w ramach Naukowej Akademickiej Sieci Komputerowej zajmujący się zdarzeniami naruszającymi bezpieczeństwo w Internecie.



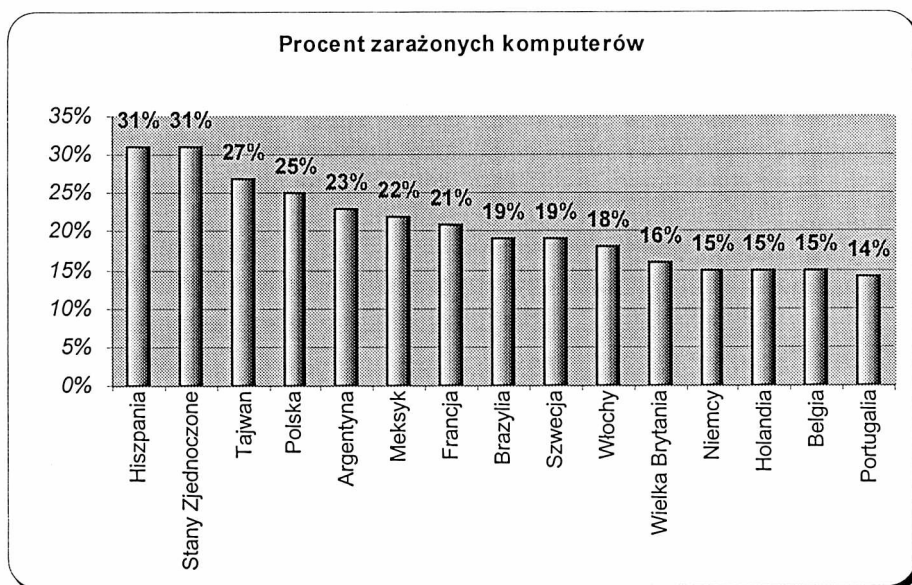
Rys. 1. Liczba incydentów notowanych przez CERT w latach 2003–2007



Rys. 2. Rozkład podtypów incydentów zarejestrowanych przez CERT Polska

Z danych zebranych przez firmę Panda Security w III kwartale 2008 roku wynika, że pod względem infekcji komputerów szkodliwym kodem Polska zajmuje czwarte miejsce na świecie (ponad 25% zarażonych komputerów przy średniej 14,48%). Uży-

skane wyniki przedstawiono na rysunku 3. Według przedłożonego raportu, 31% wszystkich wykrytych zagrożeń stanowiło adware⁴ [Bezp].



Rys. 3. Procent zainfekowanych komputerów na świecie (zaokrąglenie do 1%)

Podkreślenia wymaga zwiększona liczba zgłoszeń dotyczących kradzieży tożsamości (w ciągu ostatnich czterech lat procentowy udział tego typu zagrożeń wzrósł ponad siedmiokrotnie) oraz znaczny wzrost naruszania praw autorskich (niemal dziesięciokrotny!). W raporcie Federalnej Komisji Handlu (z grudnia 2007 roku) podano, iż straty konsumentów w ubiegłym roku na skutek różnego rodzaju oszustw szacuje się na ponad 1,2 miliarda dolarów, przy czym 32% zgłoszeń dotyczyło kradzieży tożsamości.

W październiku 2008 roku firma Symantec w raporcie „The State of Spam” podała, iż we wrześniu tegoż roku 4% wszystkich komputerów zombie⁵ działających w Internecie pochodziło z Polski. Największą liczbę zdalnie przejętych komputerów zanotowano w Turcji (12%), Brazylii (9%) i Rosji (8%). Stanowi to około 100-procentowy wzrost zjawiska w porównaniu z poprzednim miesiącem. Zdalna kontrola nad przejętym systemem pozwala go wykorzystać do prowadzenia bezprawnych działań, takich jak rozsyłanie spamu bądź propagowanie nielegalnych treści, przechowywanie skradzionych danych, czy do przeprowadzenia ataku DoS. Jednocześnie eksperci firmy Symantec zaobserwowali zwiększenie się liczby niechcianych przesyłek zawierają-

⁴ Programy wyświetlające reklamy zawierające moduł szpiegujący.

⁵ Komputery, nad którymi przejęto kontrolę, wykonujące zadania bez wiedzy ich właściciela.

cych szkodliwe oprogramowanie (ukryte w kodzie źródłowym wiadomości lub w postaci załączników). Dynamika wzrostowa jest znacząca, gdyż od czerwca do września 2008 roku procent szkodliwych kodów wykrytych w przesyłanym spamie wzrósł dwunastokrotnie.

Z najnowszych badań przeprowadzonych przez Ipsos dla AVG Technologies wynika, że w Europie ofiarą ataku padło ok. 22% komputerów ankietowanych użytkowników, a najbardziej zagrożeni byli Włosi (32%) i Brytyjczycy (31%).

Najnowsze statystyki wykazują wzrost liczby ataków skierowanych na instytucje finansowe. Jednocześnie podkreśla się narastającą liczbę ataków o charakterze kryminalnym. Do najczęściej stosowanych w tym celu technik ataku należą:

- phishing – czyli, wyłudzenie poufnych danych osobowych za pomocą dołączanych do listu poczty elektronicznej odnośników do fałszywych stron internetowych (głównie instytucji finansowych), przy czym stosowane przez przestępców metody stają się coraz bardziej wyrafinowane,
- przekierowywanie strumienia danych dzięki modyfikacji pliku „hosts” systemu Windows lub parametrów konfiguracyjnych serwera DNS,
- „man-in-the-middle” – przechwytywanie ruchu pomiędzy klientem a serwerem (między klientem a instytucją finansową) przez pośrednika.

Przestępcy nieustannie poszukują nowych metod. Ostatnio pojawiły się dwie nowe odmiany phishingu: mishing i vishing. Pierwsza odnosi się do użytkowników urządzeń mobilnych i jej celem są różnego typu urządzenia przenośne (głównie telefony komórkowe), natomiast vishing (nazwa powstała z połączenia słów voice i phishing) oznacza kradzież danych osobowych z wykorzystaniem telefonii VoIP⁶ (*Voice over IP*).

Na czele listy dziesięciu najbardziej popularnych szkodliwych kodów, opublikowanej przez Kaspersky Lab w październiku 2008 roku, znalazły się:

- koń trojański Trojan-Downloader.WMA.Wimad.n – plik multimedialny wykorzystujący lukę w odtwarzaczu Windows Media Player,
- robak Worm.Win32.AutoRun.dui,
- wirus Virus.Win32.Sality.aa.

Lista została sporządzona na podstawie liczby komputerów, na których szkodliwe programy zostały wykryte.

Laboratorium RSA Fraud Action Research Lab doniosło z kolei o trzyletniej nie wykrytej działalności w sieci niebezpiecznego konia trojańskiego Sinowal (znanego też jako Torping czy Mebroot), który zainfekował na świecie ponad 500 tysięcy komputerów, w tym także systemy instytucji finansowych. Szkodliwe oprogramowanie wykradało identyfikatory i hasła kont bankowych oraz inne informacje dotyczące kart kredytowych. Przedstawiciel RSA ujawnił, że ofiarami tego konia trojańskiego zostały duże organizacje, a włamań dokonano w Ameryce Północnej i Południowej, Azji, krajach Pacyfiku oraz Europie (także w Polsce).

⁶ Technologia transmisji głosu w sieci IP.

Nowe zagrożenie dla użytkowników Internetu stworzyło pojawienie się wirusów (np. Gpcode) szyfrujących pliki (takich aplikacji jak MS Word, Excel, Adobe Reader, czy pliki graficzne) na zainfekowanym komputerze. Za odtworzenie zawartości przestępcy żądają okupu. Pierwszy raz robak Gpcode pojawił się dwa lata temu, jednak wówczas firmie Kaspersky udało się rozszyfrować stosowany w algorytmie RSA klucz. Obecnie jego długość została zwiększona do 1024 bitów, co znacznie utrudnia rozwiązanie problemu.

3. ATAKI NA SERWERY I APLIKACJE INTERNETOWE

Bezpośrednim celem ataków najczęściej są serwery i aplikacje internetowe. W czerwcu 2008 roku firma F-Secure podała do wiadomości, że według jej danych, ofiarami ataków zostało około dwa i pół miliona serwerów. Próbę skutecznego ataku poprzedza zwykle wstępna penetracja systemu, która pozwala atakującemu między innymi na zidentyfikowanie kont użytkowników, rozpoznanie współużytkowanych plików oraz zainstalowanych aplikacji. Do pozyskiwania takich informacji służą specjalizowane narzędzia (niekiedy dostępne w Internecie), a techniki ich gromadzenia zależą od rodzaju platformy systemowej. W większości przypadków kontrolowanie sieci polega na podsłuchiwanie ruchu w celu przejęcia poufnych informacji, czasem strumień danych jest do nieuprawnionych odbiorców przekierowywany [MenŻuk 2008].

Do przeprowadzenia ataku na serwery i aplikacje internetowe często wykorzystywane są wady oprogramowania. Luki znajduje się nie tylko w systemach operacyjnych, ich komponentach i programach użytkowych, ale także protokołach sieciowych, a nawet w oprogramowaniu ochronnym. Wady wynikające z błędów programistycznych mogą na przykład służyć do przejęcia za pomocą exploitów⁷ kontroli nad zdalnym systemem. Wyróżnia się wiele typów luk w aplikacjach internetowych, między innymi są to [McCScaKur 2006]:

- pliki przykładów,
- ujawnienie kodu źródłowego,
- sprowadzenie do postaci kanonicznej,
- rozszerzenia serwera czy zatwierdzanie danych wejściowych (np. przepełnienia bufora).

Pełną ich listę zawiera projekt zespołu OWASP⁸ (*Open Web Application Security Project*). Luki istnieją także w zabezpieczeniach sieci. Choć nie są one tak liczne jak w systemach, stopień związanego z nimi zagrożenia również narasta.

⁷ Specjalnie przygotowane do tego celu kody.

⁸ Projekt zespołu wypracowującego ogólnodostępne zestawy procedur i standardów pozwalających programistom i projektantom na tworzenie bezpiecznych aplikacji webowych w różnych środowiskach.

Od połowy lat 90. poważne zagrożenie stanowią ataki z przepełnieniem bufora, które pozwalają przejąć nad systemem całkowitą kontrolę. Stosunkowo łatwo jest zaatakować system wykorzystując przepełnienie stosowe⁹ lub możliwość wprowadzenia szkodliwego kodu do sterty¹⁰. Popularne jest też tzw. przepełnienie zmiennych całkowitoliczbowych [McCScaKur 2006].

Aby uchronić system przed takimi atakami należy systematycznie instalować tzw. łatki systemowe, dostarczane przez producentów. Informacje o dostarczeniu nakładek do użytkowanego oprogramowania ogłaszane są nieustannie. Przykładowo, w styczniu 2008 roku Oracle wprowadził aż 51 poprawek bezpieczeństwa usuwających luki (w tym pięć poważnych) umożliwiające przeprowadzenie zdalnych ataków. Wady znaleziono między innymi w Oracle Database, Oracle Application Server i Oracle E-Business Suite. Badania pokazują, że tylko około 10% administratorów stosuje na bieżąco poprawki firmy Oracle [Bezp]. Należy też pamiętać, że nawet „uszczelniony” system można „odbezpieczyć”.

Przy atakowaniu aplikacji internetowych stosuje się wiele tych samych technik, co przy atakach na serwery. Popularne ataki wykorzystują wyszukiwarki sieciowe, a w celu wyszukania słabych stron oprogramowania zwykle poprzedza je gruntowna analiza atakowanej aplikacji. Przy pomocy różnego typu specjalizowanych narzędzi badane są między innymi architektura i funkcjonalność programów.

Ataki na aplikacje internetowe skupiają się wokół takich obszarów, jak:

- uwierzytelnianie,
- zarządzanie sesją,
- interakcja z bazą danych,
- akceptacja danych wejściowych.

Wiele aplikacji sieciowych wykorzystuje skrypty zapytań do baz danych umieszczonych na serwerach w języku SQL. Przechwycenie lub modyfikacja tych zapytań może prowadzić do nieprzewidzianego działania aplikacji i w konsekwencji umożliwić przejęcie nad nią lub danymi kontroli.

Do najbardziej rozpowszechnionych należą ataki XSS (*Cross-Site Scripting*), których idea polega na umieszczeniu na stronie WWW lub w hiperłączy szkodliwego kodu, którym zostaje zainfekowany komputer odwiedzającego ją użytkownika. Eksperti ostrzegają przed gwałtownym wzrostem infekcji witryn internetowych szkodliwym kodem, tymczasem dla wielu użytkowników Internetu zagrożenia płynące ze stron WWW to całkiem nieznane zagadnienie. Na przykład w czerwcu 2008 roku ofiarami fali ataków typu XSS stały się popularne strony związane z tenisem takich organizacji jak ITF (*International Tennis Foundation*) i ATP (*Association of Tennis Professionals*).

⁹ Polegają na umieszczeniu szkodliwego kodu w stosie wykonawczym procesora.

¹⁰ Sterta wykorzystywana jest do dynamicznego alokowania pamięci w czasie wykonywania programu. Jeśli atakujący zmieni ustawienia uprawnień na „Access Allowed”, uzyskuje nieuprawniony dostęp do systemu lub usługi.

Jedną z metod przeprowadzania ataków polega na wstawianiu kodu SSI (*Server Side Includes*) do pól stanowiących dla serwera dokument HTML, co umożliwia zdalną realizację poleceń. Mechanizm SSI zapewnia interaktywność i funkcjonowanie w czasie rzeczywistym bez programowania, i często jest używany do sprawdzania czasu systemowego lub do lokalnego wykonania rozkazów.

Do zaatakowania wybranego systemu można też wykorzystać luki w oprogramowaniu po stronie użytkownika (np. w przeglądarce internetowej lub innym oprogramowaniu przetwarzającym dane stron WWW, programach do obsługi poczty elektronicznej czy komunikatorach internetowych). Wyszukiwarki internetowe mogą być na przykład użyte do wskazania stron WWW podatnych na infekcję.

W czerwcu 2008 roku podano do wiadomości, że załatano luki w usłudze pocztowej Yahoo Mail. Problem pozwalający na włamanie techniką XSS dotyczył klienckiej wersji programu (w.9) oraz komunikatora internetowego Yahoo Messenger.

Od dawna wiele problemów stwarzają tzw. kontrolki ActiveX, aplikacje zapewniające obsługę różnych funkcji (jak np. odtwarzanie plików dźwiękowych) oraz programy napisane w Javie, bo w jej modułach nieustannie wykrywane są kolejne luki. Zagrożenie stwarza też JavaScript – popularny język pisania skryptów, gdyż łatwo można go wykorzystać do przeprowadzenia ataku (np. napisany w nim kod może przekierować użytkownika na zainfekowaną stronę WWW).

Do identyfikowania luk w oprogramowaniu serwerów (tj. miejsc umożliwiających przeprowadzenie ataku) służą skanery bezpieczeństwa, przeznaczone odpowiednio do ochrony hostów i sieci. Narzędzia te wskazują również sposoby ochrony systemu przed wykrytymi zagrożeniami. Skanery hostów wspomagają administratorów między innymi w definiowaniu polityki bezpieczeństwa i kontrolowaniu przestrzegania jej zasad. Umożliwiają na przykład tworzenie raportów wykazujących odstępstwa od przyjętej polityki, w których jednocześnie znajdują się sugestie sposobów rozwiązania wykrytego problemu. Natomiast skanery sieci testują sieć na obecność znanych luk i wykonują pozorowane ataki sprawdzając skuteczność wprowadzonych zabezpieczeń.

Narzędzia pozwalające testować oprogramowanie służą administratorom do badania bezpieczeństwa serwerów i aplikacji internetowych. Pozwalają na znalezienie luki umożliwiającej na przykład wywołanie przepełnienia bufora lub zablokowanie usługi. Z tych samych powodów chętnie też po nie sięgają hakerzy.

4. REALNE ZAGROŻENIA BEZPIECZEŃSTWA SYSTEMÓW SIECI ROZLEGŁYCH

Producenci systemów starają się jak najszybciej usuwać zidentyfikowane wady, jednak nieustannie pojawiają się informacje o nowo znalezionych lukach.

W marcu 2008 roku poinformowano o wykryciu dziesięciu luk (w tym trzech o znaczeniu krytycznym dla funkcjonowania danego oprogramowania) w popularnej

przeglądarki Firefox i innych produktach firmy Mozilla. Wykryte błędy między innymi powodują niestabilność pracy silnika przeglądarki, w skrajnym przypadku przerwanie jej działania (co można wykorzystać do wykonania dowolnego kodu), umożliwiają podglądnięcie historii odwiedzanych stron internetowych oraz wstawienie kodu JavaScript innej stronie WWW. W czerwcu 2008 roku podano do wiadomości, że kilka luk na swoich stronach „załatało” Google. Jedną z nich pozwalała na utworzenie witryny podszywającej się pod stronę z domeny google.com, która przekierowywała użytkowników na serwery przestępców. Jednocześnie podano informację, że Google pracuje jeszcze nad znalezioną luką w serwisie DoubleClick. Kolejną wersję swojej przeglądarki internetowej (w. 9.61) opublikowała niedawno Opera Software. Usunięto w niej błąd umożliwiający wykorzystanie funkcjonalności „history search” do zdalnego uruchomienia kodu. Niestety, w tej najnowszej wersji również znaleziono wadę (funkcja przeszukiwania historii nie weryfikuje danych wprowadzanych przez użytkownika), która stwarza to samo zagrożenie. Zatem przed firmą ponownie staje zadanie jak najszybszego wprowadzenia poprawki.

Gwarancję bezpieczeństwa sytemu stanowi bieżąca aktualizacja wykorzystywanego oprogramowania, tymczasem użytkownicy, z różnych przyczyn, nie są w stanie instalować najnowszych wersji aplikacji natychmiast po ich ukazaniu się. Z badań specjalistów do spraw bezpieczeństwa wynika, że na przykład aż 40% użytkowników Internetu narażonych jest na atak z powodu nie stosowania zaktualizowanych wersji przeglądarek sieciowych. W przypadku najczęściej wykorzystywanego Internet Explorera jest to 48%, dla Firefoxa 8%, dla Opery 10%, a dla Safari 30%, przy czym w badaniu nie brano pod uwagę rozszerzeń dla przeglądarek.

W czerwcu 2008 roku pojawiła się informacja o opublikowaniu przez firmę Adobe nowych wersji Readera i Acrobat (w.8.1.3), w których usunięto krytyczne wady wykryte w edycjach 8.1.2 i wcześniejszych. Błędy w obsłudze skryptów JavaScript pozwalają na zdalne przejęcie funkcji administratora i wykonanie kodu. Podano też informację o wykryciu nowej luki w zabezpieczeniach programu Adobe Flash, która może być wykorzystana do dystrybucji szkodliwego oprogramowania, umożliwiając infekcję dużej liczby komputerów. Przestępcy rozpowszechniają niebezpieczne pliki .swf (rozszerzenie nazwy plików programu Flash) mogące przybierać postać animacji (którą użytkownik musi uruchomić) lub obrazu ładowanego w chwili otwarcia strony WWW. Laboratorium Panda Security przewiduje w najbliższym czasie wzrost infekcji przy pomocy tej luki, gdyż oprogramowanie współpracuje z różnymi przeglądarkami. Do tej pory wykryto korzystającego z tej metody konia trojańskiego Wow.UB.

W 2008 roku ujawniono też fakt wykrycia krytycznych luk w popularnym programie graficznym Adobe PageMake (w. 7.0.1 i 7.0.2), które można wykorzystać (atak przepełnienia bufora) do przełamania zabezpieczeń komputera. Nie wyklucza się występowania tych samych wad w innych edycjach oprogramowania. W związku z licznymi kłopotami firmy Adobe w ostatnim okresie, można zastanawiać się, ile jeszcze luk znajduje się w dostarczanych przez nią narzędziach.

W październiku 2008 roku podano do publicznej wiadomości, iż PandaLabs wykryło kilka plików wykorzystujących najnowszą lukę w systemach operacyjnych firmy Microsoft (Windows 2000, XP, Server 2003), o której niedawno firma informowała (MS08-067). Szkodliwe kody umożliwiają kradzież poufnych danych (między innymi danych uwierzytelniających, na przykład haseł do komunikatorów internetowych) lub przejęcie pełnej kontroli nad komputerem zainfekowanego nimi użytkownika (np. koń trojański Gimmiv.A).

W połowie 2008 roku poinformowano o odkryciu łatwego w użyciu zestawu narzędzi umożliwiających zamianę każdego pliku wykonywalnego w robaka. Zdaniem specjalistów, aplikacja niebezpieczna staje się dopiero w rękach hakera, natomiast działania niedoświadczonych włamywaczy pozwalają jedynie odwrócić uwagę od prawdziwie przestępczych ataków.

Często słaby punkt systemu informatycznego – jak donosi European Network and Information Security Agency, która przebadła kilkaset firm we Francji, Niemczech i Wielkiej Brytanii – stanowią urządzenia podłączone do sieci (np. drukarka), które również mogą być wykorzystane do przeprowadzenia zdalnego ataku.

Zagrożenie niosą też same protokoły sieciowe. W październiku 2008 roku doniesiono o odkryciu luki w protokole TCP (dotyczy wszystkich implementacji), która umożliwia przeprowadzenie na szeroką skalę ataku DoS. Możliwość podjęcia takiego ataku jest konsekwencją otwarcia odpowiednio dużej liczby połączeń, które nigdy nie są zamykane. Sposób na ich nawiązanie wykryło dwóch szwedzkich specjalistów podczas przeprowadzania testów penetracyjnych przy pomocy narzędzia Unicorn Scan. Na ataki nie są odporne także tzw. bezpieczne połączenia, jak SSH czy SSL, w których wykorzystuje się szyfrowanie dla ochrony transmitowanych danych.

Przestępcy internetowi coraz częściej stosują atak XSS wykorzystując legalne strony WWW. Firma ScanSafe podała, iż 68% zablokowanego w maju 2008 roku szkodliwego oprogramowania używało do rozprzestrzeniania się zaufane witryny sieciowe. Stanowi to ponad 400-procentowy wzrost tego zjawiska w stosunku do maja roku ubiegłego. Ta metoda infekowania systemów jest bardziej podstępna i trudniejsza do wykrycia niż atak bezpośredni. Przykładowo, w czerwcu 2008 roku podano informację, że hakerzy z Turcji wykorzystali stronę internetową poświęconą marsjańskiej misji sondy Phoenix Mars Lander (umieszczonej na serwerach uniwersytetu Arizony) do przekierowania jej użytkowników na stronę przestępców.

Niestety, aktualizacje nie zawsze niosą długotrwałą poprawę bezpieczeństwa. Na przykład w listopadzie 2008 roku podano informację o pojawieniu się nowego robaka (Wecorl¹¹, MS08-067.g¹²) wykorzystującego lukę „załataną” miesiąc wcześniej przez firmę Microsoft. Okazuje się, że w tym przypadku hakerom wystarczyło kilka dni od czasu wydania przez Microsoft poprawki, by zmodyfikować istniejący wcze-

¹¹ Oznaczenie firmy Symantec.

¹² Oznaczenie używane przez firmy Microsoft i Kaspersky Lab.

śniej szkodliwy kod (który zmusił firmę do wydania poprawki) i stworzyć nowe zagrożenie. Działalność robaka sprowadza się do zainstalowania konia trojańskiego (Trojan-Dropper.Win32.Agent.yhi) oraz mającego go maskować rootkita¹³ (rootkit.Win32.KernelBot.dg). Zainstalowany na jednym komputerze będzie się starał zarazić pozostałe komputery danej sieci lokalnej, co jest szczególnie groźne dla firm mających ochronę jedynie przed zagrożeniami z zewnątrz.

5. FAŁSZYWE NAKŁADKI SYSTEMOWE I PROGRAMY OCHRONNE

W ostatnim okresie pojawiło się nowe zagrożenie, wymagające odrębnej uwagi, jakim jest podsuwanie użytkownikom fałszywych poprawek systemowych i programów ochronnych. Przestępcy internetowi różnymi drogami (np. poprzez maile) rozpowszechniają informacje o plikach zawierających rzekome poprawki firmy Microsoft do systemu operacyjnego Windows. Listy podobnej treści przygotowywane są z dbałością o szczegóły, co czyni je wiarygodnymi.

Według firmy G. Data Software¹⁴ należy szczególnie uważać na zainfekowaną aktualizację programu Adobe Flash Player, bowiem wyszukiwarka internetowa między innymi udostępnia (na wysokiej pozycji), zamiast rzekomej aktualizacji, link sfalszowanej strony zawierającej plik z wirusem z rodziny Win32:Agent.

Alarmujące doniesienia płyną ze strony firmy Panda Security, z której badań wynika, iż gwałtownie rośnie liczba infekcji wywołanych przez fałszywe programy antywirusowe. Pobrane z sieci fałszywe oprogramowanie, po uruchomieniu, komunikuje fakt znalezienia w systemie szkodliwych kodów (wcześniej mogą pojawić się na ekranie różnego typu animacje sugerujące infekcję lub awarię systemu) i powiadamia użytkownika o konieczności zakupu płatnej wersji oprogramowania. Szacuje się, że problem ten dotyczy już ponad 30 milionów komputerów. Wobec takiego zagrożenia użytkownicy Internetu są bezradni. Nie wszyscy są w stanie problem rozpoznać, a całkowite usunięcie szkodliwego oprogramowania jest żmudne i może zająć kilka dni nawet doświadczonemu osobom. Laboratorium PandaLabs wykryło dotąd ponad 7 tysięcy odmian fałszywych programów antywirusowych. W październiku 2008 roku specjalista do spraw bezpieczeństwa w Panda Security Polska podał do wiadomości, że według danych firmy, 3% użytkowników podczas dokonywania zakupu fałszywego narzędzia było zmuszonych podać swoje dane osobowe, co stanowiło dla nich dodatkowe zagrożenie [Bezp]. Rynek fałszywego oprogramowania zabezpieczającego analizowała również firma SecureWorks. Z jej badań wynika, że najbardziej rozpowszechnionymi fałszywy-

¹³ Specjalizowane, pomocnicze we włamaniach narzędzie służące do ukrywania niebezpiecznych plików i procesów umożliwiających utrzymanie kontroli nad systemem operacyjnym.

¹⁴ Jeden z producentów oprogramowania zabezpieczającego.

mi programami antywirusowymi są Antivirus XP 2008 oraz Antivirus XP 2009 sprzedawane przez rosyjską firmę Bakasoftware za pośrednictwem osób rekrutowanych na nielegalnych forach internetowych. Szacuje się, że na fałszywych programach antywirusowych przestępcy zarabiają łącznie miliony dolarów rocznie.

W obecnej dobie do ochrony systemów informatycznych należy wykorzystywać profesjonalne wielofunkcyjne narzędzia zabezpieczające, bowiem programy antywirusowe i personalne zapory sieciowe już dawno przestały być wystarczające.

7. PODSUMOWANIE

Systemy sieci rozległych, w tym rozproszone systemy bazodanowe, nieustannie narażone są na różnego rodzaju zagrożenia sieciowe. Specjaliści firmy Microsoft donoszą wprawdzie o spadku liczby wykrywanych luk (w pierwszej połowie 2008 roku było ich w oprogramowaniu tej firmy o 19% mniej, niż rok wcześniej), jednak podkreślają coraz większe zaawansowanie techniczne podejmowanych ataków i szkodliwych kodów. Z tego powodu przed administratorami sieci i systemów staje niezwykle trudne zadanie skutecznej ochrony danych i realizowanych w sieci usług. Do analizy podatności serwerów internetowych na ataki wykorzystuje się specjalizowane narzędzia umożliwiające ocenę stanu ich bezpieczeństwa. W połowie 2008 roku firmy Microsoft i HP zaprezentowały darmowe narzędzia wspomagające zabezpieczanie stron internetowych przed powszechnymi atakami XSS i SQL injection, których celem jest identyfikacja luk w oprogramowaniu i minimalizacja skutków potencjalnego ataku. Także firma Google pospieszyła z pomocą swoim użytkownikom, i tym właścicielom serwisów internetowych, którzy zapiszą się do serwisu Google Webmaster Tools, będzie wysyłała ostrzeżenia w przypadku, gdy będą korzystać z przestarzałego, podatnego na atak oprogramowania sieciowego.

Stworzenie rozwiązań zapewniających wysoki poziom ochrony systemów teleinformatycznych, a w szczególności gwarantujących bezpieczną realizację transakcji on-line, stanowi problem o najwyższym priorytecie.

LITERATURA

- [Bezp] <http://bezpieczenstwo.onet.pl>
- [Cert] www.cert.pl
- [McCScaKur 2006]] McClure S., Scambray J., Kurtz G., *Hacking zdemaskowany. Bezpieczeństwo sieci – sekrety i rozwiązania*. Wydawnictwo Naukowe PWN S.A., Warszawa 2006.
- [MenŻuk 2008] Mendyk-Krajewska T., Żukowski G., *Podsluchiwanie ruchu sieciowego w aspekcie bezpieczeństwa systemu teleinformatycznego*. W: *Współczesne aspekty sieci komputerowych*. Praca zbiorowa pod red. P. Gaja, B. Pochopienia i S. Kozielskiego, WKŁ, Gliwice 2008.

- [RusMul 2004] Rusell R., Mullen T., Dan F.X. i inni, *Hakerzy atakują. Jak przejąć kontrolę nad siecią*, Helion, Gliwice 2004.
- [SchGreCar 2002] Schetina E., Green K., Carlson J., *Bezpieczeństwo w sieci*, Wydawnictwo Helion, Gliwice 2002.
- [SchMen 2004] Scheliga M., Mendrala D., *Bezpieczeństwo Twojego komputera*, Helion, Gliwice 2004.
- [Sym] www.symantec.pl
- [Wik] www.pl.wikipedia.org

THREATS OF WIDE AREA NETWORK SYSTEMS – SIZE OF PROBLEM

This paper concerns the problem of IT system threats in wide area network. The problem of threats in computer network has been analyzed by many specialized institutions which systematically publish effects of their studies. At present common viruses and worms appear in network rarely, while adware, spyware and Trojans occur more frequently. Hackers use defects in a system software and applications for criminal purposes. They benefit from this procedures financially. Producers correct faulty software as quickly as possible, but new defects are found permanently. False system and applications updates and false antivirus computer programs are a new kind of threat.

*social network, actors, connection, security,
Internet networking, data control, privacy, web 2.0.*

Dmitrij ŻATUCHIN*

SOCIAL PYRAMID METHOD IN SOCIAL NETWORK SERVICES

Recently social networks in Internet became something more than just discussion lists, instant messaging and websites, where we can share our hobbies or communicate to meet new pals. They started to unite network population, in more and more specific networks, called social networks with areas of interest or culturally bounded networks (for ex. LinkedIn, Migente, Xing, GoldenLine). At that moment such networks, which bring specific purpose of existence and goals of membership are called to be social networks 3.0 [Hornik]. But during analysis of possibilities for actor, or member in such network it was concluded, that he can't control his attendance and information share with other members more than in binary way – $\{0, 1\}$.

In this paper was introduced well-known theory first used in ancient society of Egyptians – social pyramid, and show how it may be applied to Internet social networking. With the help of such pyramid, actor receives powerful tool for control his level of closeness of connection with other actors in networks. It's being told about "ultimate console for consumer control" and suggested to use such feature in advance [Young 2006]. Social pyramid also could be used as a gate for information in such console, what have been mentioned at the end.

1. INTRODUCTION

Social Pyramid as a social study was described by historians relating to ancient Egypt, where definite division by social class was dictated by an Egyptian's profession. Such social stratification can be wrapped in a pyramid (Figure 1). People also believed that the gods gave them their positions in society.

At the bottom of the "Social Pyramid" were slaves, those who were captured in war had totally no rights. Then, on the next levels, stand soldiers, farmers, and tomb builders – people, who represented the greatest percent of the Egyptian population. The workers supported the professionals above them, just as the base of every building supports the rest. Above the workers were skilled craftsmen, such as artists, merchants, usual doctors. Next in the level were government officials, Pharaoh's soldiers

* Instytut Informatyki, Wydział Informatyki i Zarządzania Politechniki Wrocławskiej, 50-370 Wrocław, Wybrzeże Wyspiańskiego 27, dmitrij.zatuchin@pwr.wroc.pl

and scribes. Of course, at the top of pyramid was the pharaoh – god and ruler for whole society. What is curious, that in ancient times women had more rights than in half of 20th century – for example they could buy and sell property, however they did not take part in government business.

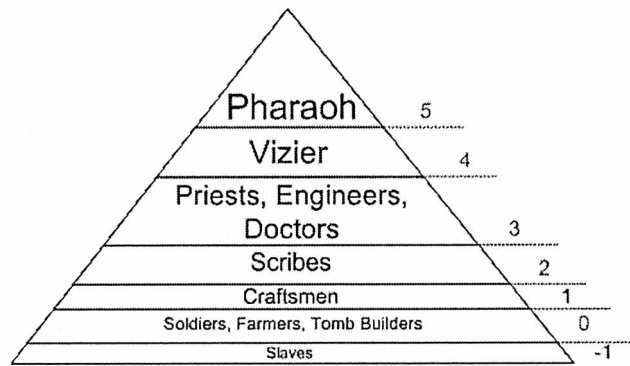


Fig. 1. Social pyramid in ancient Egypt

2. SOCIAL PYRAMID IN SOCIAL NETWORKS

At the beginning there should be reminded “1% rule”. Taking a group of 100 representatives, only one of them will create content for Web, 10 will interact with it, by writing comments or offering improvements and the rest will just browse it [Horowitz]. Web 2.0 social network sites are finding the same thing. The whole creative web is rather small group of contributors, which attract the commentators or editors, which in turn attracts the huge audience. Such phenomenon can be observed in blogs, forums or especially Wikipedia.org. Below figure 2 shows the division of Internet population.

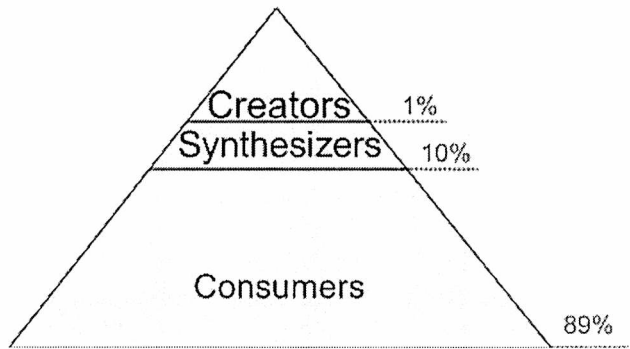


Fig. 2. Internet population pyramid

2.1. INFORMATION ACCESS – PRESENT SITUATION

Majority of Internet community accesses information in a scheme showed on Figure 3 – users of websites accept or decline visitors, show them information about themselves only after registration or invitation to friends list. More technically, it is a binary situation with some minor states of 0 and 1, when owner of information accepts to share his profile data to actors or everyone in system, otherwise make it private – only actors, who know exact URI or verified actors (users on a contact list, in the same private social network) can view it. Some users can feel uncomfortable with such state-of-art of privacy control in social network website.

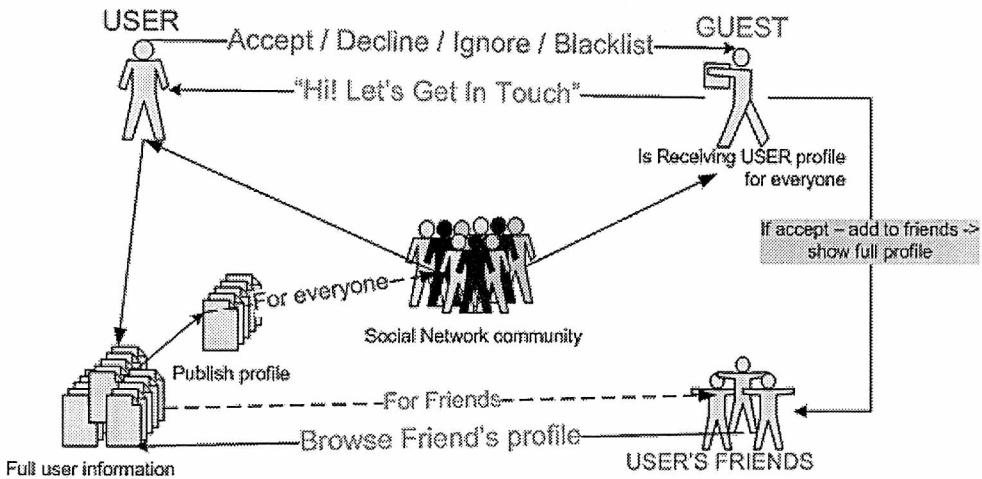


Fig. 3. Accessing content in social networks

2.2. MODIFIED RULE FOR RELATIONSHIP

If we try to copy social pyramid model created for real life to Internet community, it would look like shown on Figure 4.

Second pyramid on every level includes people added to set of actual, by copying them from higher levels. Why that is so? It's just an assumption, that someone who is close to us with the weight of 3, is also close to us with the weight factor = 2, the only exception is blacklist, which is under social pyramid. Let's define some basic attributes of pyramid. Pyramid level $p_l(u_{id})$ is a factor for closeness of actors, which create user's private population in a social network; visibility level $v_l(u_{id})$ is the default level of publicity of information we provide for actors in social network, $owner_pyramid$ – set of user identification attributes, which state that they are in pyramid relationship.

$p_l(u_id) \in N \cup \{-1\}$, where
 u_id – user's identification attribute,
 $\{-1\}$ is a state for blacklist – set of users below information access,
 $v_l(u_id) \in N \cup \{-1\} | v_l(u_id) \leq p_l(u_id)$

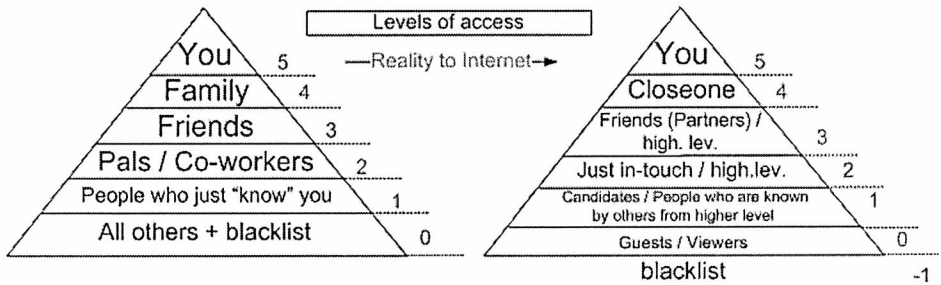


Fig. 4. Social pyramid – transformation to Internet

Activities, done by owners and actors, determine the way how social pyramid between them will look like on Figure 5.

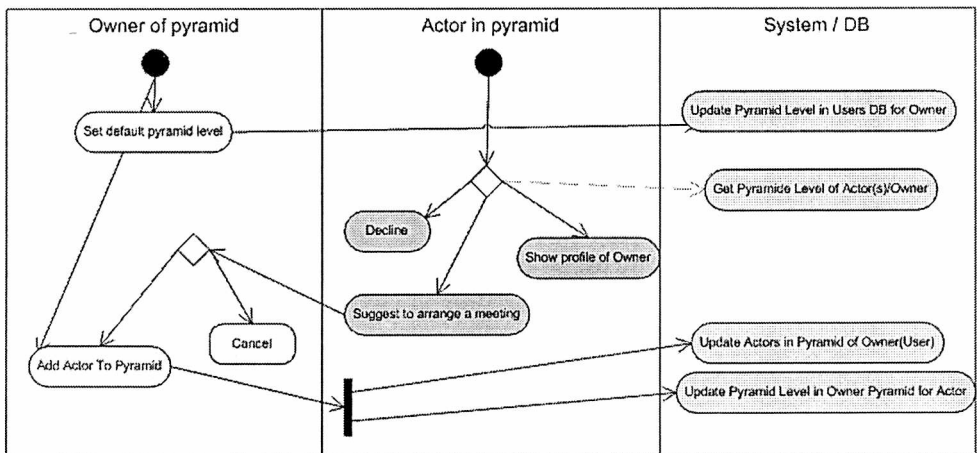


Fig. 5. Activity diagram for pyramid in social networks

Three main rules for pyramid have been described in pseudo language:

a) Rule of class minority, pyramid shows owner's information if:

$p_l(owner_id) \leq (GET\ v_l(actor_id)\ WHERE\ actor_id \in owner_pyramid)$

b) Rule of suggestion:

$IF\ p_l(owner_id) > (GET\ p_l(actor_id)\ FROM\ owner_pyramid)\ OR$
 $actor_id \in (ANY\ OF\ actor2_pyramid\ WHERE\ actor2_id \in owner_pyramid))$
 $AND\ p_l(actor_id) \geq p_l(owner_id)$

c) Rule of decline:

*IF (p_l(actor_id) WHERE actor_id @owner_pyramid)<0 OR NOT
(actor_id @ (actor2_pyramid WHERE actor2_id @ owner_pyramid)).*

3. PRACTICAL USAGE OF SOCIAL PYRAMIDS

Below there is shown an example of database model and sample queries for basic tasks.

3.1. DATABASE MODEL

Model of pyramid for Users2 table, which keeps information about social network users in database, has been prepared on Figure 6. Logical and optimal relationship was achieved by usage of foreign key.

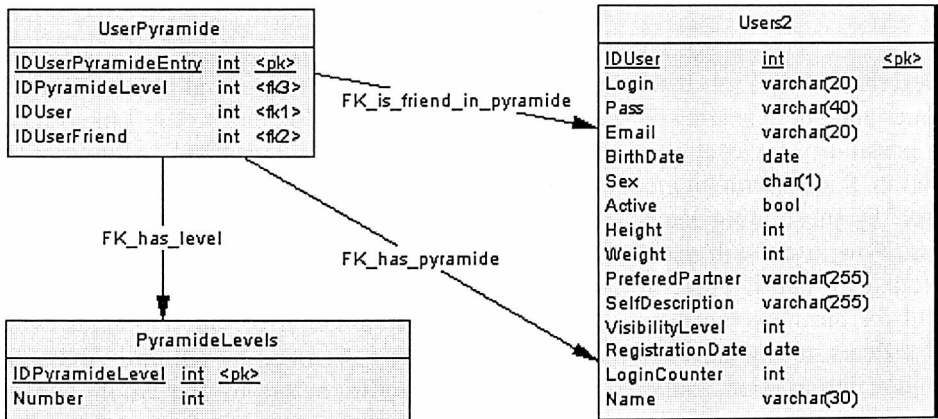


Fig. 6. Social pyramid – database model

Table *PyramideLevels* fulfils the need of user granulation.

3.2. SQL QUERIES

Sample SQL queries are provided for above database model of social pyramid:

a) Addition of User's Friend to contact group with weight equal 5

INSERT INTO UserPyramide(IDUser,IDUserFriend,IDPyramideLevel) values(300,100,(SELECT IDPyramideLevel FROM PyramideLevels WHERE Number=5));

- b) Show all friends of friends with level in pyramid equal 2
SELECT Login FROM Users JOIN UserPyramide ON UserPyramide.IDUserFriend=Users.IDUser WHERE UserPyramide.IDUser=300 AND IDPyramideLevel=(SELECT IDPyramideLevel FROM PyramideLevels WHERE Number=2);
- c) Sample “promotion” of friend to higher level in pyramid
UPDATE UserPyramide SET IDPyramideLevel=(SELECT IDPyramideLevel FROM PyramideLevels WHERE Number=1) WHERE IDUser=100 AND IDUserFriend=300;
- d) *UPDATE Users SET VisibilityLevel=2 WHERE IDUser=300;*
- e) *DELETE FROM UserPyramide WHERE IDUser=300 AND IDUser-Friend=100;*

Using such queries, server language interactions are provided for example in PHP or JSP, by checking *Number* attribute of *User* with the appropriate of *Friend*, which is different from *User*.

4. SUMMARY

Described model was discovered in process of commenting social network services for dates and photo sharing. The lack of control was felt. Somehow only LinkedIn.com gave possibility for control of user profile – it could be viewed only after accepted invitation. Social pyramid model could be also transformed in an “ultimate console for control” by substituting actors’ data information with information, to which they have access. Now, users will decide, if to show and publish something interesting they’ve just found for public view, or keep it to them. Moreover, such tool can be very successful for business social networks, such as LinkedIn.com or Goldenline.pl, where chief officers, technicians and managers can create subgroups by naturally managing their joint information, gathered on stage of collecting actors to one whole sub network, if to make it public for other people, not from branch, not from current company, or be free, innovative, open-minded and bring it for everyone. Everything depends on view, though information will be still in hands of possessors. Of course, in every solution there are always advantages and negatives, and one of them is narrowing access of other users to information giving instead more privacy for another.

REFERENCES

- [Atwood] Atwood J., Scoble R., *Value in Networking*, LinkedIn Corporation, <http://linkedin.com>
- [Horowitz] Horowitz B., *Creators, Synthesizers, and Consumer*, <http://www.elatable.com/blog>
- [Hornik] Hornik B., *Social Networks 3.0*. <http://ventureblog.com>
- [Young 2006] Young R., *Can MySpace be beaten?*, <http://gigaom.com/2006/02/26/can-myspace-be-beaten>

METODA PIRAMIDY SPOŁECZNEJ W SERWISACH SPOŁECZNOŚCIOWYCH

Ostatnio sieci społeczne w Internecie stały się czymś więcej niż tylko listami dyskusyjnymi, czatami i stronami internetowymi, gdzie możemy podzielić się informacjami o naszym hobby lub komunikować się po to by np. poznać nowych znajomych. Sieci te zaczęły jednoczyć społeczności w grupy coraz bardziej ukierunkowane na konkretne obszary zainteresowań lub z narzuconymi ograniczeniami kulturowo-obyczajowymi (np. LinkedIn, Migente, Xing, Goldenline). Obecnie sieci, które wprowadzają określony cel istnienia i cele związane z członkostwem nazywane są sieciami społecznymi 3.0 [Hornik]. Jednak analizując funkcjonalności dostępne uczestnikom w takiej sieci stwierdzono, że mogą oni kontrolować dostęp do informacji i sposób jej współdzielenia nie więcej niż w sposób binarny 0 lub 1.

W niniejszej pracy przedstawiono, znany z czasów społeczeństwa starożytnych Egipcjan, model piramidy społecznej i pokazano, w jaki sposób może on być zastosowany w zarządzaniu danymi w sieciach społecznych w Internecie. Za pomocą takiej piramidy użytkownik otrzymuje potężne narzędzie kontroli poziomu jego bliskości połączenia z innymi podmiotami w sieci.

Dmitrij ŻATUCHIN*

BAZY DANYCH W TESTOWANIU UŻYTECZNOŚCI STRON WWW

Użyteczność angażuje wiedzę z zakresu psychologii, fizjologii, anatomii, inżynierii, projektowania i zarządzania. Można ją testować automatycznie, badać z użytkownikami, analizować kontekst użycia poszczególnych funkcjonalności. Niestety, wyniki, które są uzyskiwane podczas badań metodą empiryczną czy analityczną, są nieusystematyzowane – znajdują się w luźnych notatkach, plikach tekstowych czy arkuszach kalkulacyjnych, co rzutuje na wydłużenie cyklu życia projektu.

Badania użyteczności stron WWW (ang. *Web usability*) cechuje pewna powtarzalność czynności, podobieństwo definiowanych zadań dla uczestników testu, zbierane funkcjonalności czy prototypy, których testowanie mogłoby zostać przyspieszone dzięki zastosowaniu baz danych ze specyfikacją produktów, uczestników badań, przydzielanych zadań oraz wyników. Pozwoli to na kategoryzację i przeszukiwanie informacji dotyczących poszczególnych testów, co wpłynie na zwiększenie efektywności pracy zespołów projektowych i testerów.

1. WPROWADZENIE

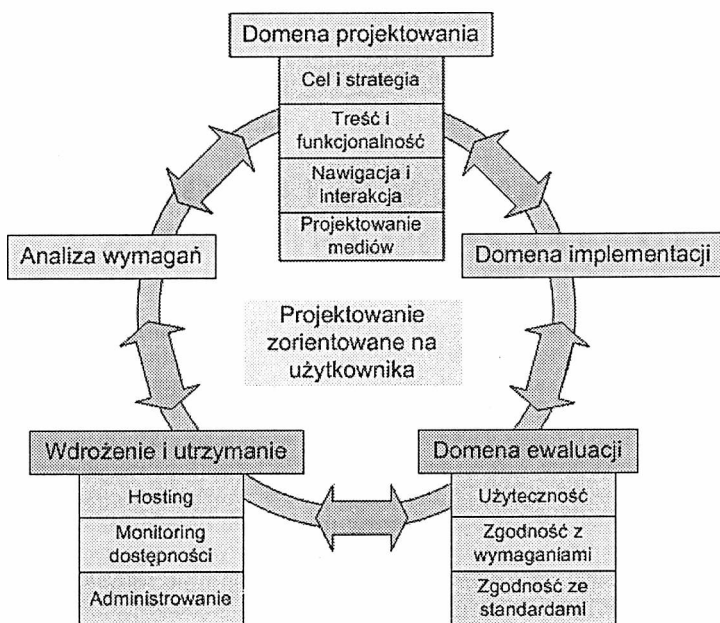
Mówiąc o testowaniu użyteczności produktu należy zwrócić uwagę na definicję samej użyteczności. Według standardu ISO 9241 (część 11, punkt 3.1) użytecznością nazywamy stopień, w jakim produkt może zostać użyty przez określonych użytkowników, aby osiągnąć określone cele w sposób:

- skuteczny, co oznacza dokładność i kompletność, z jaką użytkownik osiąga cel,
- efektywny, co oznacza, że przeznaczone zasoby wystarczą na skuteczne osiągnięcie celu (np. czas),
- satysfakcjonujący, co oznacza brak negatywnych emocji, swobodę w używaniu produktu w określonym kontekście użycia.

Z kolei według IEEE (ang. *Institute of Electrical and Electronics Engineers* – Instytut Inżynierów Elektryków i Elektroników) – Standardu 610.1, użyteczność jest to łatwość, z jaką użytkownik uczy się operować, wprowadzać dane oraz interpretować wyniki systemu lub komponentu.

* Instytut Informatyki, Wydział Informatyki i Zarządzania Politechniki Wrocławskiej, 50-370 Wrocław, Wybrzeże Wyspiańskiego 27, dmitrij.zatuchin@pwr.wroc.pl

Użyteczność powinna stanowić szkielet procesu projektowania. Kontekst użycia w tym wypadku przesądza o cechach systemu, zaś zadania, które realizują użytkownicy, mają znaczny wpływ na architekturę produktu podczas wytwarzania oprogramowania. Na rysunku 1 przedstawiono cykl życia projektu, należy zwrócić uwagę na domenę ewaluacji.



Rys. 1. Projektowanie zorientowane na użytkownika. Model cyklu życia projektu

Projekt jest wiązany z zadaniami, klasami użytkowników i kontekstem użycia, powstaje przy wykorzystaniu standardów, jako źródła wiedzy poprawnego projektowania, np. ISO 9241, standardów WCAG (ang. *Web Content Accessibility Guidelines*) czy wskazówek takich jak Sun Java Look&Feel, Gnome 2.0, Windows Guidelines.

Badanie użyteczności w Internecie ma na celu zweryfikowanie takich elementów procesu projektowania jak:

- organizacja struktury strony (układy kolumnowe),
- nazewnictwo sekcji, odnośników, kontrolek,
- rozmieszczenie informacji (problem intuicyjnego odnalezienia),
- grupowanie elementów,
- porządek zdarzeń (konsekwentny ciąg akcji i oczekiwania użytkowników co do wyniku ich akcji).

Dostarczając produkt do ewaluacji z użytkownikiem, projektant waliduje założenia i wychwytyje błędy, które mają różne stopnie skomplikowania – dosyć łatwo jest poprawić etykietę kontrolki, zaś problem z nielogiczną strukturą lub niewłaściwym działaniem funkcjonalności może wymagać przejrzenia początkowych założeń i skut-

kować ponownym zaangażowaniem zasobów na przeprojektowanie i wdrożenie. Dlatego bardzo ważne jest uwzględnienie użyteczności na wczesnych etapach projektowania (analizy, modelowania i prototypowania) produktu.

2. EWALUACJA STRON WWW

Ewaluacja według nowego modelu referencyjnego ISO/AWI 23973 *Software ergonomics for World Wide Web user interfaces* (pol. Ergonomia oprogramowania dla interfejsów WWW) jest jedną z domen procesu projektowania. Zawiera ona narzędzia i techniki używane po to, aby ocenić ostateczny projekt, np. poprzez testy użyteczności, co wspiera podejmowanie decyzji i stosowaniu odpowiednich wskazówek w następujących obszarach:

- **Cel i strategia.** Jaki jest cel strony i w jaki sposób zostanie to przekazane użytkownikowi?
- **Zawartość i funkcjonalność.** Jaki jest model konceptualny strony? W jaki sposób będzie prezentowana treść, jak rozwiązana będzie personalizacja i prywatność?
- **Nawigacja i interakcja.** Jaka powinna być architektura informacji? W jaki sposób rozwiązana kwestia nawigacji – umieszczenia głównego menu, pomocniczego, odnośników? W jaki sposób użytkownik będzie przeszukiwać stronę?
- **Prezentacja i projekt mediów.** Jak będą wyglądać poszczególne strony, aby użytkownicy mogli pożytecznie wykorzystać informację prezentowaną na nich? Jak powinny być używane multimedia?

Wszak ewaluacja jest pojęciem bardziej ogólnym, odnoszącym się do sprawdzania i weryfikowania produktu stosując wybraną metodę.

2.1. METODY BADAŃ

Metody ewaluacji, w szczególności, stron internetowych, klasyfikuje się następująco [Żatuchin 2008]:

1. Empiryczne testy:

- grupy fokusowe,
- ankiety,
- wywiady (w tym samowyywiad zwany inaczej testem Kruga [Krug 2005]),
- listy kontrolne,
- testy z użytkownikami:
 - zorientowane na użytkownika:
 - ocena prototypu papierowego,
 - testy doświadczalne.
 - obserwacyjne (obserwacja pracy z systemem),
 - otwarta eksploracja.

2. Analityczne testy:

- GOMS (ang. *Goal, Operations, Methods, Selection rules*),
- model odkrywczego uczenia się (ang. *cognitive walkthrough*),
- metody heurystyczne (na przykład Jacob Nielsen [Nielsen 1994]) i eksperckie (analiza specjalisty),
- testy symulacyjne (wykonywanie określonych scenariuszy i przechodzenie przez ścieżki serwisu internetowego).
- automatyczne testy bezinwazyjne (śledzenie ruchów, kliknięć i akcji).

2.2. ZWROTNE DANE

Przeprowadzając badania i testy pozyskuje się wiele danych, które wymagają odpowiedniego przetworzenia. Na podstawie otrzymanych danych opracowuje się raporty [Kuniavsky 2003].

W zależności od użytych metod otrzymywane są następujące rodzaje danych:

- notatka obserwatora dotycząca przebiegu ewaluacji z użytkownikami,
- lista kontrolna z odpowiedziami,
- uporządkowany zbiór danych ze skończoną liczbą pól, uzupełniany w trakcie badania w bazie danych lub arkuszu kalkulacyjnym przez prowadzącego badania lub obserwatora,
- wywiad przedtestowy i potestowy w postaci ankiety,
- dane pozyskane automatycznie za pomocą sprzętu (na przykład Eyetrack 6000) lub oprogramowania (na przykład GTAnaly, CrazyEgg, Noldus The Observer):
 - ślad skupienia wzroku (ang. *gaze trail*),
 - obszary patrzenia (ang. *look zones*),
 - analiza konturów (ang. *contour analysis*),
 - mapa cieplna (ang. *heat map*),
 - akcje wykonywane przez użytkownika (ang. *actiontracking*).
- mapa kliknięć (ang. *click map*) i przestojów (ang. *noop points*):
 - pozycja kursora, czas, rodzaj akcji,
 - średni czas wykonywania każdego zadania oraz średnia liczba kliknięć w element o przydzielonym typie,
 - rozkład uczestników kończących zadania,
 - ślad użytkownika – uporządkowany w czasie zbiór punktów od startowego do końcowego.
 - parametry użytkownika (przeglądarka, rozdzielczość, lokalizacja GeoIP, źródło wejścia na stronę, strona wyjścia).
- analiza ekspercka:
 - analiza heurystyczna (skończony zbiór pytań i odpowiedzi),
 - analiza porównawcza konkurencji (skończony zbiór cech).

- dane metryczne:
 - liczba kliknięć potrzebna do osiągnięcia celu,
 - czas potrzebny do osiągnięcia celu.

W przypadku testów z użytkownikami występuje zależność liczby użytkowników i skuteczności rezultatu. Wykazano [Nielsen 1993], iż przeprowadzenie badań z 8 użytkownikami skutkuje wykryciem około 95% błędów, zaś przy pięciu użytkownikach wynik jest o kilkanaście punktów gorszy i wynosi około 82%.

Aby zwiększyć efektywność poszukiwania rozwiązań problemów występujących w serwisach WWW oraz wyeliminować powtórzenia testów, wystarczy zgromadzić odpowiednio wyważoną bazę z ewaluacji różnych stron internetowych w jeden zbiór, w którym będą zawierać się instancje modeli danych dla poszczególnych testów, na przykład dla testu z użytkownikiem składającą się ze zbioru zapytań, obserwacji i wyników, czy w przypadku motiontrackingu (śledzeniu ruchów) model metody zakłada, że pozyskiwana jest następująca czwórka danych $MT = \{P, T, L, A\}$, gdzie [Zatuchin 2008]:

$P = (x, y)$, jednostką dla x, y jest 1px (1 piksel).

$T = 00:00:00$, h:m:s.

$L = 000$ ms – wyrażone w milisekundach.

$A \in \{\text{NULL, START, STOP, CLICK, FIX, CLOSE}\}$.

Istnieją również testy, których wynik jest binarny, między innymi badanie zgodności kodu poprzez walidatory HTML/CSS/WAI czy testy kontrastu kolorów na stronie (algorytm LCR). Takie testy można traktować jako atrybuty badanej strony WWW.

Jak można zauważyć, powstaje zbiór danych nieusystematyzowanych, niespójnych i rozproszonych. Jednym z rozwiązań jest utworzenie czterech głównych baz danych, połączonych w centralnym systemie wspierającym prace nad użytecznością produktu.[Butler 2003] Są to:

1. Baza specyfikacji produktu.
2. Baza wyników z procesu testowania.
3. Baza potencjalnych kandydatów do testów i testerów systemu.
4. Baza narzędzi pomocnych w ewaluacji.

Nie jest to optymalne rozwiązanie. W przypadku testowania stron internetowych bardziej logiczne jest utworzenie bazy danych stron internetowych (systemów) i powiązanych z nimi użytkownikami, multimediami i wystąpieniami modeli testów, powiązanych w relacji N:N.

2.3. RODZAJE DANYCH

Podczas ewaluacji stron uzyskiwane są dane, które po przeniesieniu do komputera można podzielić na:

- tekstowe (pliki .txt, .doc, .xls),
- graficzne (zrzuty ekranów, zdjęcia, skany szkiców),
- multimedialne (wywiady w postaci nagrań audio, video z kamer).

Ponadto, w czasie automatycznych testów wykonywanych poprzez metodę click-tracking-u, eyetracking-u czy walidatory zwracane są duże ilości informacji w postaci plików (Tabela 1), wpisów do oddzielnych baz danych czy sesji zapisanej po stronie przeglądarki lub serwera badanego użytkownika.

Tabela 1. Dane uzyskiwane podczas automatycznego testowania

System clicktracking	486 87 998 opera 1 1200958706460 82.143.157.23 592 95 998 opera 1 1200958709735 82.143.157.23 1122 238 1257 firefox 1 1200958981668 62.21.16.7 599 91 1257 opera 2 1200962745875 83.12.143.166
Eyetrack6000	Czytodziala.pl – Mozilla Firefox, F, 57, 337, 604, 20.658, 20.698, 0.040, 0, Czytodziala.pl – Mozilla Firefox, F, 58, 206, 604, 20.718, 20.858, 0.140, 0, Czytodziala.pl – Mozilla Firefox, F, 60, 822, 603, 21.098, 21.358, 0.260, 0, Czytodziala.pl – Mozilla Firefox, F, 61, 795, 604, 21.518, 21.638, 0.120, 0

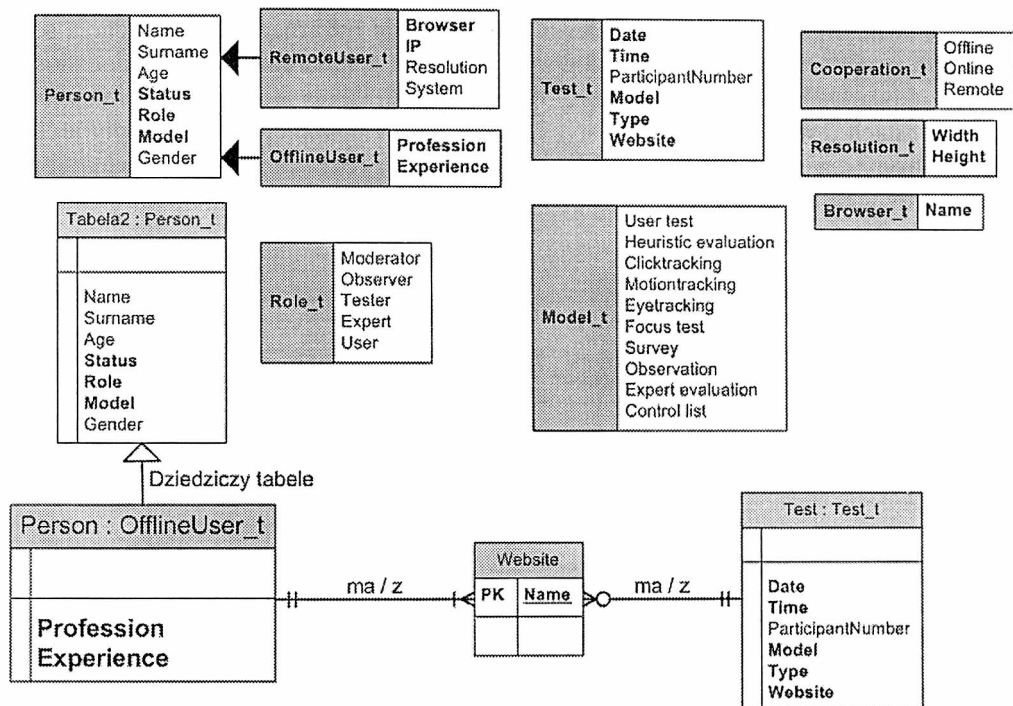
3. MODEL OBIEKTU BADANIA UŻYTECZNOŚCI

Aby zacząć budować bazę danych wspomagającą ewaluację stron WWW, należy zacząć od zdefiniowania modelu obiektu samego badania oraz jego uczestników. W testach mogą uczestniczyć użytkownicy, obserwatorzy, eksperci, moderatorzy, czyli osoby. Stąd powstaje klasa abstrakcyjna typu Osoba, która może być dziedziczona przez konkretny typ uczestnika badania. Osoba może mieć takie ogólne atrybuty wynikające najczęściej z ankiety uczestnika (dane osobowe, wiek, doświadczenie w pracy z komputerem/Internetem, wykształcenie) oraz z rodzaju testu (inwazyjny, bezinwazyjny, zdalny, lokalny, w środowisku natywnym, obcym itp.).

3.1. OBIEKT „OSOBA”

Obiekt osoby jest rozbudowany i wymaga objęcia szerokiej grupy osób, które biorą udział w badaniach użyteczności. Na rysunku 2 został przedstawiony ogólny model obiektowo-relacyjny bazy danych systemu wspomagającego ewaluację stron WWW. Wyróżnia on podstawowe cechy typu osoby, wyszczególnia role, takie jak moderator, obserwator, tester, ekspert i użytkownik, ponadto informuje o typie współpracy – zdalnym, na miejscu z komputerem czy bez. Przykładowo zostały wyszczególnione dwa typy – *RemoteUser_t* (zdalny użytkownik) oraz *OfflineUser_t*

(uczestnik testu offline). Warto zwrócić uwagę na powiązane ze sobą encje – *Person*, *Website* i *Test*.



Rys. 2. Schemat przedstawiający występujące typy danych i związki między podstawowymi obiektami

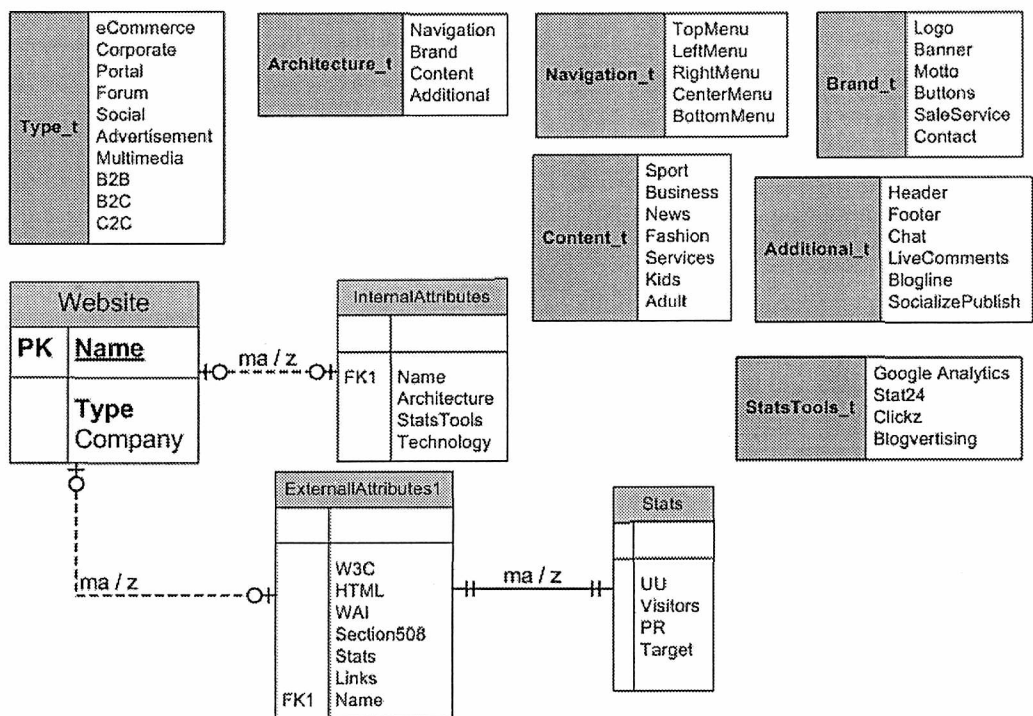
Każdy obiekt typu *Model_t* reprezentować będzie odpowiednią metodę ewaluacji zastosowaną do danej strony WWW i powinien zostać zdefiniowany osobno, gdyż wymaga ustanowienia związku z pozostałymi obiektami, między innymi obiektem wyników i obiektem z danymi zbieranymi podczas ewaluacji.

Test typu *Test_t* jest ogólną definicją modelu każdego testu, który odbywa się w określonym czasie, z określoną skończoną liczbą osób, należy do jednego lub kilku rodzajów modelu typu *Model_t* oraz jest związane ze stroną internetową, której obiekt został zdefiniowany w następnym podrozdziale.

3.2. OBIEKT „STRONA”

Obiekt strony internetowej jest kluczowy w całym modelu badania użyteczności, gdyż to jest centralny punkt samego testowania. W zależności od cech wejściowych

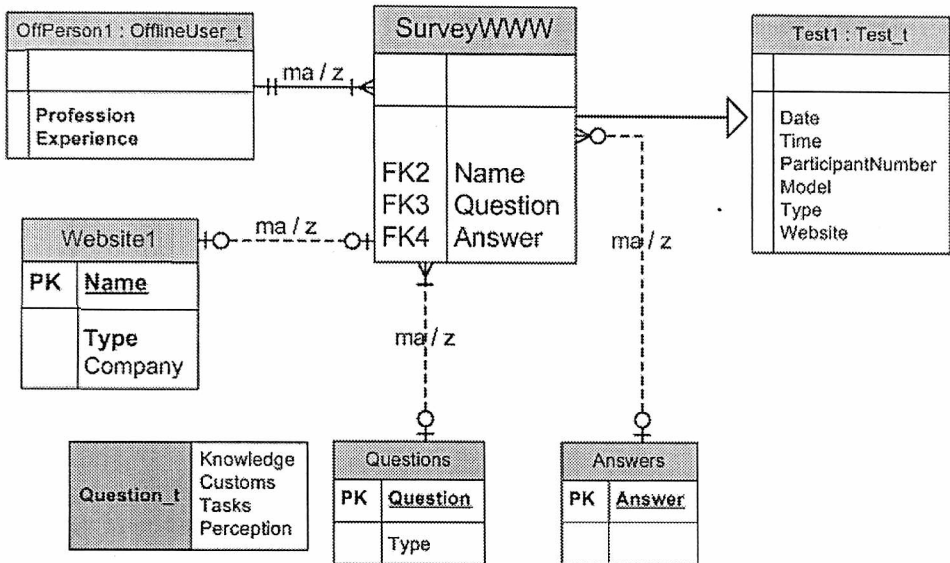
strony i wymagań zewnętrznych podejmowana jest decyzja o wybraniu odpowiedniej metody ewaluacji. Na rysunku 3 przedstawiono obiektowo-relacyjny model strony internetowej. Encja *InternalAttributes* dotyczy architektury informacyjnej i cech opisujących stronę. Encja *ExternalAttributes* zawiera takie informacje jak walidacja serwisu pod względem dostępności, zgodności kodu ze standardami oraz widoczności w wyszukiwarkach. Można zauważyć w *InternalAttributes* kolumnę *Architecture*, która w testach jest wiązana ze statystykami kliknięć w poszczególne elementy architektury strony czy np. encją ankiety dotyczącą opinii o rozmieszczeniu informacji na stronie. Powinna też powstać encja *OtherWebsites*, która jest listą indeksów stron podobnych badanej strony.



Rys. 3. Typy danych w modelu strony (Website) oraz powiązania obiektu Website

3.3. ZWIĄZEK OSOBA-TEST-STRONA

Na rysunku 4 pokazano model ankiety *SurveyWWW*, w którym zawarte są informacje o użytkowniku i wrażeniach po udziale w badaniu. Obrazuje on związek testu, osoby i strony WWW.



Rys. 4. Model ankiety strony (SurveyWWW)

4. PODSUMOWANIE

Celem pracy było pokazanie istnienia możliwości zwiększenia efektywności prowadzenia testów użyteczności poprzez zastosowanie obiektowo-relacyjnego modelu bazy danych. Dla każdej metody prowadzenia badania użyteczności strony internetowej wymagane jest utworzenie odpowiedniej relacji i obiektów. Wstępne prace pokazują, że liczba obiektów, ich typów i relacji może być bardzo duża i baza danych od strony technicznej może wprowadzać komplikacje w proces przebiegu badań, w szczególności, gdy użytkownikami będą osoby o małym stopniu zaawansowaniu pracy z systemami bazodanowymi. Z drugiej strony pozwoliłoby to na usystematyzowanie wiedzy pozyskiwanej w trakcie takich badań oraz stworzyłoby repozytorium plików opisanych wiedzą w bazie danych. Należy stworzyć intuicyjny system, zaprojektowany zgodnie z metodologią user-centered-design (rys. 1), który wykorzystałby taką bazę i tym samym ułatwił pracę projektantów, testerów i ekspertów.

LITERATURA

- [Butler 2003] Butler M., Lahti E., *Lotus Notes Database Support for Usability Testing*, Lotus Development Corporation, 2003.
- [Krug 2005] Krug S., *Don't Make Me Think: A Common Sense Approach to Web Usability*, New Riders Press, 2005, 20–30, 168–180.

- [Kuniavsky 2003] Kuniavsky M., *Observing The User Experience – A Practitioner's Guide To User Research*, Morgan Kaufmann Publishing, 2003.
- [Nielsen 1993] Nielsen J., Landauer, T.K., *A mathematical model of the finding of usability problems*, Proceedings of INTERCHI 93, 206–213, New York, NY: ACM, 1993.
- [Nielsen 1994] Nielsen J., *Usability Engineering*, Boston, MA: Academic Press Professional, 1994.
- [Zatuchin 2008] Żatuchin D., *Automatyzacja procesu badania użyteczności stron WWW*, Praca magisterska, Politechnika Wroclawska, Wrocław 2008.

DATABASES IN USABILITY EVALUATION OF WEBSITES

Usability involves knowledge of psychology, physiology, anatomy, engineering, design and management. It can be tested automatically, with users, analyze the context of the use of the specific functionality. Unfortunately, the results that are obtained in an empirical or analytical evaluation, are not systemized – loose notes, text files and spreadsheets, which impinge on the extension of the life cycle of the project.

Tests of websites are characterized by a repeatability of operations, similar to the tasks defined for the test participants, concurrent functionality or prototypes, testing which could be accelerated through the use of databases with the specifications of the products involved in research, allocation of tasks and results. It allows to search and categorize information on individual tests, which will lead to increased efficiency of project teams and testers.

Wydawnictwa Politechniki Wrocławskiej
są do nabycia w księgarni
„Tech”
plac Grunwaldzki 13, 50-377 Wrocław
budynek D-1 PWr., tel. (0-71) 320-29-35
Prowadzimy sprzedaż wysyłkową

ISBN 978-83-7493-443-5